

DEPARTMENT OF MATHEMATICS AND STATISTICS  
COLLEGE OF SCIENCES  
OLD DOMINION UNIVERSITY  
NORFOLK, VIRGINIA 23529

**AUTOMATION OF REVERSE ENGINEERING  
PROCESS IN AIRCRAFT MODELING AND  
RELATED OPTIMIZATION PROBLEMS**

By

W. Li, Co-Principle Investigator

and

J. Swetits, Co-Principle Investigator

Progress Report

For the period ended December 1994

Prepared for

National Aeronautic and Space Administration

Langley Research Center

Hampton, VA 23665

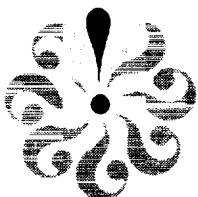
Under

**Cooperative Agreement NCC1-68 Supplement 15**

Dr. Robert E. Smith Jr., Technical Monitor

SID - Scientific Application Branch

November 1994



## FOREWORD

This is the progress report on the research project "Surface Modeling and Optimization Studies of Aerodynamic Configurations." Within the guidelines of the project, our attention was directed toward research activities in the area of surface reconstruction by optimization techniques and spline smoothing. The period of performance of this specific research was from January 1, 1994 to December 31, 1994. Our research was supported by the NASA Langley Research Center through Cooperative Agreement NCC1-68. The cooperative agreement was monitored by Dr. Robert E. Smith Jr. of System and Information Division (Scientific Application Branch), NASA Langley Research Center.

# **AUTOMATION OF REVERSE ENGINEERING PROCESS IN AIRCRAFT MODELING AND RELATED OPTIMIZATION PROBLEMS**

Dr. W. Li and Dr. J. Swetits  
Old Dominion University

## **Summary**

During the year of 1994, we studied reverse engineering problems in aircraft modeling. Our initial concern was to obtain a surface model with desirable geometric characteristics. Much of the effort during the first half of the year was to find an efficient way for solving a computationally difficult optimization model. Since the smoothing technique in the proposal "Surface Modeling and Optimization Studies of Aerodynamic Configurations" requires solutions of a sequence of large-scale quadratic programming problems, it is important to design algorithms that can solve each quadratic program in a few iterations. Our research led to 3 papers by Dr. W. Li, which were submitted to SIAM Journal on Optimization and Mathematical Programming. Two of these papers have been accepted for publication.

Even though significant progress has been made during this phase of research and computation time was reduced from 30 minutes to 2 minutes for a sample problem, it was not good enough for on-line processing of digitized data points. After discussion with Dr. Robert E. Smith Jr., we decided not to enforce shape constraints in order to simplify the model. As a consequence, we adopted P. Dierckx's nonparametric spline fitting approach, where one has only one control parameter for the fitting process – the error tolerance. At the same time we also tested the surface modeling software developed by Imageware. Our research indicates a substantially improved fitting of digitized data points can be achieved if a proper parameterization of the spline surface is chosen. A winning strategy is to incorporate Dierckx's surface fitting with a natural parameterization for aircraft parts.

The report consists of 4 chapters. Chapter 1 provides an overview of reverse engineering related to aircraft modeling and some preliminary findings of our effort in the second half of the year. Chapters 2-4 are the research results by Dr. W. Li on penalty functions and conjugate gradient methods for quadratic programming problems.

# TABLE OF CONTENTS

FOREWORD .....	i
SUMMARY .....	ii
TABLE OF CONTENTS .....	iii
Chapter 1	
Automation of Reverse Engineering Process in Aircraft Modeling .....	1
1.1 Introduction .....	1
1.2 Issues in Surface Reconstruction .....	3
1.3 Description of Research Results .....	4
1.4 Future Research Initiatives .....	6
Bibliography .....	9
Chapter 2	
Differentiable Piecewise Quadratic Exact Penalty Functions for Quadratic Programs with Simple Bound Constraints .....	11
2.1 Introduction .....	12
2.2 Unconstrained Reformulations .....	16
2.3 Second Order Optimality Conditions .....	22
2.4 Equivalence of The Quadratic Program and Its Unconstrained Reformulation .....	25
2.5 Conclusions .....	31
Bibliography .....	33
Chapter 3	
A Conjugate Gradient Method for Strictly Convex Quadratic Programs with Simple Bound Constraints .....	39
3.1 Introduction .....	39

3.2 Conjugate Gradient Methods for Convex Quadratic Splines .....	44
3.3 Strictly Convex Quadratic Programs with Simple Bound Constraints .....	50
3.4 Conclusions .....	54
Bibliography .....	56
Chapter 4	
Differentiable Exact Penalty Functions via Hestenes-Powell-Rockafellar's Augmented Lagrangian Function .....	59
4.1 Introduction .....	59
4.2 Some Properties of the Augmented Lagrangian Function .....	63
4.3 Exact Penalty Function in Primal Variables .....	67
4.4 Exact Penalty Function in Dual Variables .....	73
4.5 Comments .....	76
Bibliography .....	78

# Chapter 1

## Automation of Reverse Engineering Process in Aircraft Modeling

In this chapter we analyze the reverse engineering process in aircraft modeling and point out what can make the reverse engineering process more accurate and efficient.

### 1.1 Introduction

Reverse engineering is a relatively new terminology emerging from late 80's [3]–[15]. If one researches the literature on reverse engineering, one will find references on reverse engineering in software development and maintenance as well as reverse engineering related to computer-aided manufacturing. Perhaps it is appropriate to call it reverse mechanical engineering when it is related to modeling of a mechanical part. Mechanical engineering is a process that starts with a design and ends with a mechanical product. Therefore, any process that starts with a mechanical part and ends with a design might be called reverse mechanical engineering. Recently, with the development of advanced laser digitizer and computer-aided manufacturing, reverse mechanical engineering is not only a research initiative but also an industrial reality. Many researchers are looking for ways of building a “gigantic me-

chanical copying machine" which automatically produces a replica of any given mechanical part (or its scaled model).

Reverse mechanical engineering has many applications, such as manufacturing of a part from a new physical model, replication of a mechanical part that does not have a computer-recognizable design, analysis/modification of a mechanical part that does not have a computer-recognizable design, and verification of a mechanical part that has a computer-recognizable design. For aircraft modeling, reverse engineering can be used for verification of the reliability/accuracy of a scaled-down model based on a computer design and CFD analysis of the structure of a new aircraft model based on its spline surface representation.

In general, a reverse (mechanical) engineering process consists of 5 procedures: (1) acquisition of data points (it is done by using some coordinate measuring machine, most likely a laser digitizer), (2) separation of data (it is related to pattern recognition in artificial intelligence, such as extracting the data points on a wing of an aircraft), or feature identification of the model, (3) surface fitting of each part that has a simple geometric structure, (4) reassembling of parts by using CAD programs, (5) computer-aided manufacturing (CAM) of the original model based on the mathematical model.

The important issues related to reverse engineering process in aircraft modeling are accuracy and efficiency. With enough resources, one can create spline surface patches from digitized data points that represent the original model. However, there are a few sources of inaccuracy in the modeling process: (1) imperfection of the original model, (2) error in measurement by a laser digitizer, (3) missing data points, and (4) error occurred in reconstruction of surface patches based on the digitized data points. The first three types of errors are not our primary concern at the moment. As a matter of fact, they are relatively insignificant when compared with the human error in reconstruction of surface patches. In order to produce an acceptable surface model, we have to reduce fitting errors in surface reconstruction. With advanced digitizing technology, it takes a few seconds to measure millions of data points on a surface. How to process a huge amount of data points becomes a critical issue in surface reconstruction. Therefore, in order to make the reverse engineering process work, we have to address accuracy and efficiency issues in reconstruction of spline surface patches from digitized data points.

## 1.2 Issues in Surface Reconstruction

The reconstruction of spline surface patches from digitized data points can be roughly described as the following problem: given a set of measured data points  $\{(x_i, y_i, z_i) : 1 \leq i \leq n\}$  on an aircraft, find spline surface patches that represent the aircraft "accurately" and "efficiently". This seemingly naive mathematical problem is actually very difficult and complicated (cf. [3]). It involves many different fields in mathematics, such as Computer-Aided Geometric Design (CAGD), Approximation Theory, Numerical Optimization Techniques, Nonparametric Regression in Statistics, and Numerical Linear Algebra (for software coding).

First let us consider the accuracy issue. Ideally, one wants to find a spline surface such that all the measured points are on the surface. Then we know that, at least at those measured points, the spline surface faithfully represents the aircraft model. However, there are several reasons for us not to use such a spline surface: (1) due to imperfection of the aircraft model and/or measurement error, the measured point positions are not accurate and such a spline surface normally has local oscillatory structure which is not realistic as an aircraft's surface; (2) computationally, it might be impossible to find such a surface and the related mathematical problem is called spline interpolation in approximation theory; (3) even if one can find such a spline surface, it usually requires a complicated mathematical form and its representation might have tens of thousands of coefficients which is not desirable. As a consequence, one has to set an error tolerance for the spline surface representation of an aircraft model. A common practice is to use the standard deviation as a measure of accuracy of the spline surface representation. However, error analysis of fitting data points by a spline surface is a quite complicated mathematical problem. Our objective is to improve surface fitting techniques so that better surface representations can be produced as a result.

Another issue is efficiency. There are two kinds of efficiency in surface reconstruction: efficiency in use of resources and efficiency in mathematical surface representation. If one has to spend days and months in order to produce a desirable surface by using a software package, then the software is not efficient for the reconstruction of a surface. If the mathematical surface needs tens of thousands of parameters for its representation, then the representation is not efficient. Here our objective is to design programs that



can quickly produce a spline surface with only a few hundreds of parameters, even though the digitized data set might contain a quarter of a million to a million of data points.

However, there is a conflict in achieving our objectives. In general, a more accurate representation of the data points yields a more complicated spline surface which requires more parameters for its mathematical form. Sometimes, we can not improve the accuracy of a surface representation while using a simple mathematical form, and there has to be a trade-off between the accuracy of a surface approximation and the simplicity of surface representation. One might use some strategy in image compression, where the rate of compression conflicts with the quality of the compressed image. In wavelets compression of digitized images, Saito [1] used minimal information description length to determine the optimal trade-off between accuracy and simplicity. Similar ideas might be used here.

### 1.3 Description of Research Results

How can we design a program that produces a simple and accurate spline surface representation of millions of digitized data points? In this section, we describe one possible way to accomplish this task.

The reconstruction of a surface from digitized data points consists of the following processes: (1) separation of a model into geometrically simple parts, such as separation of an aircraft into wings, tails, and fuselage, etc. (this process is called data separation in pattern recognition engineering); (2) representation of each part as a spline surface (surface fitting); (3) reassembling of the spline patches as one surface that represents the original model. So far, our effort is on the surface fitting part of the modeling process. The reason is that inaccurate surface fitting results in erroneous spline surface representation of an aircraft model.

During the summer of 1994 when our research was funded by NASA through the grant NCC-1-68 Supplement-15 from NASA Langley Research Center, we studied various techniques for surface fitting. In particular, we studied a shape-control regression model by forcing constraints on derivatives, the B-spline curve/surface modeling software "SURFACER" developed by Imageware, and the curve/surface fitting package in NETLIB library designed by Dierckx. Here is a summary of our findings:

1. Even though a shape-control regression model can produce a surface with a desirable shape, it is very time consuming. The shape-control regression model by B-splines is more complicated and the related optimization problem is more difficult to solve. Therefore, we abandoned the shape-control modeling idea temporarily, since long turn-around time is not acceptable in practice. However, we do want to produce spline surfaces with certain geometric features, such as convexity or bi-convexity. Especially, the shape of wings are crucial for its aerodynamic characteristics. What we hope for is that, if the fitting of digitized data is accurate enough, then the resulting surface would resemble the actual model.
2. Our overall impression of "SURFACER" is very good. Its interface is user-friendly and one can learn how to use the package in a short time. The package is a state of art product and has great potentials. However, there are three aspects of this product which still require further research and development: (1) automatic pattern separation, (2) parameterization of spline surface, and (3) surface fitting. Right now, one has to identify each part of a model by naked eyes, even though it has some artificial intelligence features to help the user. The objective should make the interface transparent to the user so that one knows how those automatic pattern recognition features can be used. For a given set of data points, the spline surface generated by the package is heavily dependent on the boundary curves one creates for the data set. In some cases, the boundary curves completely determine the final output, which is undesirable. The problem is the choice of parameterization of the fitting spline surface. Due to the same problem, the free form surface fitting is not reliable at all as pointed out by Dr. Kurt Skifstad, the president of Imageware. Therefore, it is easy to find the 4 boundary curves and a B-spline surface by using the package, but the result might not be desirable. Then you have to go back and repeat the process again. Sometimes delicate decisions have to be made during the boundary curve fitting process so that the final spline surface would be acceptable.
3. The curve/surface fitting package designed by Dierckx [2] is perfect for aircraft modeling in the sense that there is only one control parameter:

the fitting error tolerance. If you want to fit a set of data points by a B-spline surface, then you provide a tolerance on how far you allow the spline surface to deviate from the given data set and the program does the rest: it decides how many knots are needed, where to put those knots, and finally produces a smoothest spline surface among the ones that fit the data set within the given error tolerance. We believe that this is the best strategy for aircraft modeling: finding the simplest spline surface that fits the data with an acceptable error tolerance. However, Dierckx's spline surface fitting subroutine can only fit a data set  $\{(x_i, y_i, z_i) : 1 \leq i \leq n\}$  where  $(x_i, y_i)$  are points inside a rectangular region. But, in general, one first fits the boundary of a data set by 4 boundary curves, then uses the Gordan-Coons interpolant to get a parameterization of the surface, and finally fits the data set by 3 spline surfaces  $(x(u, v), y(u, v), z(u, v))$  where  $(u, v)$  are parameters defined on a rectangular region. If we use Dierckx's curve/surface fitting package, it will enhance the performance of "SURFACER", where one has to use many different parameters to control the fitting process. For designers, more control parameters provide flexibility in design process; but, for reverse engineering, many different control parameter could be a nightmare when deciding which set of parameters would give the best fit.

## 1.4 Future Research Initiatives

Our objective is to enhance some aspects of "SURFACER", which will be purchased by NASA Langley Research Center, so that it can produce better spline fitting of digitized data points of an aircraft. Here are some ideas which can make the surface reconstruction more efficient and more accurate:

1. automation of some steps in the surface modeling process so that it takes less time to produce a desirable spline surface model,
2. a better parameterization method for the fitting splines so that more reliable and more accurate spline surfaces can be generated during the fitting process.

Note that any sound automation not only reduces the cost of producing a surface model (in terms of computer time and manpower), but also eliminates

potential human errors in the process. Therefore, automation is not only an efficiency issue but also an accuracy one. Our goal is to have a program that takes an error tolerance and 4 corner points (in the digitized data set) and produces the simplest spline surface (it can) that fits the data set within the given error tolerance. Technically, we would like to solve the following three mathematical problems:

1. Fit the boundary points by a moving spline frame so that one does not have to fit the boundary points by 4 spline curves which are bonded together by CAGD programs. The reason is that additional errors result from the so-called stitching process (in "SURFACER"). What we want is to design a program that takes an error tolerance and four corner points and produces a closed spline curve with 4 corner points (which we call a spline frame). One possible approach is to use spline curve fitting with interpolation of 4 corner points. However, due to unreliability of digitized data, it is better to allow the fitting program to decide which positions should be the corner points by using information on positions of all boundary points. In statistical terms, we want to use all boundary points to predict the corner points. This will make the program more robust.
2. Use a natural parametric space for the digitized surface. For example, it is natural to use cylindrical coordinates for fuselage, conical coordinates for the sharp front of an aircraft, and rectangular coordinates for wings. From our experiments, an appropriate choice of the parametric space can make a dramatic difference in accuracy of the fitting process and can prevent some undesirable side-effects, such as the drifting of the fitting spline surface in some coordinate direction.
3. Use inverse Gordan-Coons mapping to reformulate the fitting problem as a fitting problem on a rectangular region. The current practice is to use the Gordan-Coons mapping instead of the inverse Gordan-Coons mapping. For example, suppose that we have a set of data points  $\{(x_i, y_i, z_i) : i \leq i \leq n\}$ , where  $z_i = f(x_i, y_i)$  for some unknown function  $f(x, y)$  and  $(x_i, y_i)$  are uniformly distributed in some region  $S$ . One can imagine that  $S$  is the projection (or shadow) of a wing. By using 4 boundary curve representation of the boundary points of the given data set, one can explicitly write down the Gordan-Coons mapping that

maps  $[0, 1] \times [0, 1]$  to  $S$ . If we want to produce a surface model of the wing by the parametric equations:  $x = x(u, v), y = y(u, v), z = z(u, v)$  for  $0 \leq u \leq 1, 0 \leq v \leq 1$ , then  $z_i$  should equal  $z(u_i, v_i)$  where  $(u_i, v_i)$  should be determined by  $x_i = x(u_i, v_i)$  and  $y_i = y(u_i, v_i)$ . Note that  $x = x(u, v), y = y(u, v)$  define the Gordan-Coons mapping that maps  $[0, 1] \times [0, 1]$  to  $S$ . Therefore,  $(u_i, v_i)$  is the image of the inverse Gordan-Coons mapping. Finding  $(u_i, v_i)$  involves solving a system of two non-linear equations, which should not be too difficult. But a common practice is to use a grid net of the rectangular region  $[0, 1] \times [0, 1]$  and, for each  $(u_s, v_t)$ , find a data point  $(x_{i(s,t)}, y_{i(s,t)})$  that is closest to  $(x(u_s, v_s), y(u_s, v_t))$ . Then fit  $(u_s, v_t, z_{i(s,t)})$  by a spline surface  $z = z(u, v)$ . One can clearly see that artificial errors are introduced in this process.

# Bibliography

- [1] N. Saito, Simultaneous noise suppression and signal compression using a library of orthonormal bases and the minimum description length criterion, in "Wavelets in Geophysics", eds. E. Foufoula-Georgiou and P. Kumar, Academic Press, 1994 (to appear).
- [2] P. Dierckx, Curve and Surface Fitting With Splines, Monographs on Numerical Analysis, Oxford University Press, 1993.
- [3] S. S. Sinha, From 3D imaging to CAD: bridging the gap to growth, Advanced Imaging, November, 1992, pp. 42-45.
- [4] S. Motavalli and B. Bidanda, A part image reconstruction system for reverse engineering of design modifications, Journal of Manufacturing Systems, Vol. 10, 1991, 383-395.
- [5] F. Hansen, E. Pavlakos, and E. Hoffman, PARES: a prototyping and reverse engineering system for mechanical parts-on-demand on the national network, Journal of Manufacturing Systems, Vol. 12, 1993, 269-281.
- [6] W.-L. Kwok and P. J. Eagle, Reverse engineering: extracting CAD data from existing parts, Mechanical Engineering, Vol. 113, 1991, 52-55.
- [7] Coordinate measuring machine software facilitates reverse engineering, Modern Machine Shop, Vol. 60, Feb. 1988, pp. 188.
- [8] J. H. Kreiner and C. S. Putcha, A computer aided technique for shaft design through Monte Carlo simulation, Computers and Structures, Vol. 52, 1994, 367-372.

- [9] S. Lee, G. J. Kim, and G. A. Bekey, Combining assembly planning with redesign: an approach for more effective DFA, Proceedings IEEE International Conference on Robotics and Automation, Vol. 3, 319-325, 1994.
- [10] G. J. Kim and G. A. Bekey, Constructing design plans for DFA redesign, Proceedings IEEE International Conference on Robotics and Automation, Vol. 3, 312-318, 1994.
- [11] T. M Sobh, J. Owen, C. Jaynes, M. Dekhil, and T. C. Henderson, Industrial inspection and reverse engineering, Proceedings of the 1994 Second CAD-Based Vision Workshop, 1994, 288-235.
- [12] T. M Sobh, C. Jaynes, M. Dekhil, and T. C. Henderson, Intelligent Systems: Safety, Reliability and Maintainability, Proceedings of the NATO Advanced Research Workshop, 1994, 95-122.
- [13] V. Kalra, P. N. Sheth, and L. G. Richards, Reverse engineering of mechanical components within a computer-aided design/computer-aided manufacturing environment, Journal of Engineering Design, Vol.2, 1994, 219-230.
- [14] R. Poli, G. Coppini, and G. Valli, Recovery of 3D closed surfaces from sparse data, CVGIP: Image Understanding, Vol. 60, 1994, 1-25.
- [15] C. Bradley, G. W. Vickers, and M. Milroy, Reverse engineering of quadratic surfaces employing three-dimensional laser scanning, Proceedings of the Institute of Mechanical Engineering, Vol. 208, No. 1, 1994, p. 21.

## Chapter 2

# Differentiable Piecewise Quadratic Exact Penalty Functions for Quadratic Programs with Simple Bound Constraints

The quadratic program with simple bound constraints is reformulated as unconstrained minimization of a differentiable piecewise quadratic function. The two problems have the same set of local solutions, the same set of isolated local solutions, and the same set of global solutions. Unlike other penalty functions, a parameter involved in the unconstrained reformulation can be easily determined by the spectrum radius of the Hessian of the objective function in the original quadratic program. Moreover, the exact penalty function can also be derived from Hestenes-Powell-Rockafellar's augmented Lagrangian function for two-sided inequality constrained minimization problems by using Fletcher's multiplier function.



## 2.1 Introduction

Consider the following quadratic program with simple bound constraints:

$$\min_{l \leq x \leq u} \frac{1}{2} x^T M x - b^T x, \quad (2.1)$$

where  $M$  is an  $n \times n$  symmetric matrix,  $b \in \mathbf{R}^n$  (a vector of  $n$  components), and  $l, u$  are vectors of  $n$  components with  $l < u$ . (Note that, if  $l_i = u_i$ , then  $x_i \equiv l_i$  and one can replace  $x_i$  by  $l_i$  in (2.1) and reformulate the problem so that  $l < u$ .) Some components of  $l$  or  $u$  may be  $-\infty$  or  $+\infty$ .

One can take advantage of the special structure of (2.1) to design numerical algorithms for solving (2.1) [1, 3, 4, 5, 6, 7, 8, 9, 11, 8, 16, 9, 18, 13, 14, 22, 23, 24, 28, 19, 20, 24, 34, 19, 36, 41, 42, 21, 44, 45, 50, 51, 26]. The article by Moré and Toraldo [42] contains references on quadratic programs with simple bound constraints in engineering applications.

When  $M$  is positive semidefinite, Li and Swetits reformulated (2.1) as the following unconstrained minimization problem [24, 25]:

$$\min_{x \in \mathbf{R}^n} \Psi_0(x), \quad (2.2)$$

where  $\Psi_0(x)$  is a convex quadratic spline defined as follows:

$$\begin{aligned} \Psi_0(x) := & \frac{1}{2} x^T (E - E^2) x - x^T E h \\ & + \frac{1}{2} \| (l - (Ex + h))_+ \|^2 + \frac{1}{2} \| ((Ex + h) - u)_+ \|^2. \end{aligned} \quad (2.3)$$

Here  $\alpha$  is a positive constant such that  $0 < \alpha \|M\| < 1$ ,  $\|M\|$  is the 2-norm of the matrix  $M$ ,  $E := I - \alpha M$ , and  $h := \alpha b$ . It was proved that  $x^*$  is a solution to (2.1) if and only if  $x^*$  is a minimizer of  $\Psi_0(x)$  [24, 25]. Therefore, algorithms for unconstrained minimization of the convex quadratic spline  $\Psi_0(x)$  can be used to find a solution to (2.1).

When  $M$  is a positive definite matrix,  $\Psi_0(x)$  is actually a strictly convex quadratic spline. In this case, one can use either a Newton method with line search to find the unique solution  $x^*$  of (2.1) in finite iterations [24, 25] or a conjugate gradient method to generate a sequence of iterates which converge linearly to  $x^*$  [20]. Moreover, the conjugate gradient method finds  $x^*$  in finite iterations if  $x^*$  is nondegenerate [20]. In general, it is very easy to design a linearly convergent descent method for finding a minimizer of a convex

quadratic spline which is bounded below on  $\mathbf{R}^n$ , even if the set of minimizers of the convex quadratic spline is unbounded [19]. As a consequence, by the unconstrained reformulation (2.2), one can easily generate a sequence of iterates which converges linearly to a solution of (2.1) when  $M$  is positive semidefinite and (2.1) has a solution [19].

The unconstrained reformulation of (2.1) was solely based on the observation that  $x^*$  is a stationary point of  $\Psi_0(x)$  (i.e.,  $\Psi'_0(x^*) = 0$ ) if and only if  $x^*$  satisfies the first order optimality conditions of (2.1) (i.e.,  $(x^*, Mx^* - b)$  is a Karush-Kuhn-Tucker point of (2.1)) [24, 25]. Therefore, it is unclear whether the minimizers of  $\Psi_0(x)$  correspond to the solutions of (2.1) when  $M$  is not positive semidefinite. In order to establish the correspondence without the convexity assumption, we have to relate the second order optimality conditions of (2.1) to those of (2.2).

When  $0 < 2\alpha\|M\| < 1$ , based on careful manipulations of the second order optimality conditions and an observation that  $\Psi_0(x)$  is actually the sum of the objective function of (2.1) and a quadratic spline penalty term, we prove that  $x^*$  is a local solution (or an isolated local solution) of (2.1) if and only if  $x^*$  is a local minimizer (or an isolated local minimizer) of  $\Psi_0(x)$ . Moreover,  $x^*$  is a global solution of (2.1) if and only if  $x^*$  is a global minimizer of  $\Psi_0(x)$ . In other words, the unconstrained reformulation (2.2) is "absolutely" equivalent to (2.1) when  $0 < 2\alpha\|M\| < 1$ . We want to point out that  $\Psi_0(x)$  can also be derived from the Hestenes-Powell-Rockafellar's augmented Lagrangian function for two-sided inequality constraints by using Fletcher's multiplier function. See Section 2 for details.

In general, one can use exact penalty functions or exact augmented Lagrangian functions to derive unconstrained reformulations of a constrained minimization problem (cf. [3, 13]). Consider the following constrained minimization problem:

$$\min_{x \in X} f(x), \quad (2.4)$$

where  $X \subset \mathbf{R}^n$  is the feasible set defined by some linear or nonlinear constraints. In order to reformulate (2.4) as an unconstrained minimization problem, we introduce an exact penalty function  $F(x, \epsilon)$  which contains a penalty parameter  $\epsilon$  and consider the following unconstrained minimization problem:

$$\min_{x \in \mathbf{R}^n} F(x, \epsilon). \quad (2.5)$$

The most important property of an exact penalty function is that its local (or global) minimizers are local (or global) solutions of the original constrained minimization problem, which is the main focus of most of the literature on this subject as noted by Han and Mangasarian [16]. This property ensures that any unconstrained minimization technique for finding a local (or global) minimizer of the penalty function yields a local (or global) solution of the original constrained problem. Usually, there is a threshold  $\epsilon^*$  for the penalty parameter such that  $F(x, \epsilon)$  is exact if  $0 < \epsilon \leq \epsilon^*$ . However, in almost all cases, there is no simple way to compute the value of  $\epsilon^*$ . Also, as pointed out by Di Pillo and Grippo [13], in practice, the threshold  $\epsilon^*$  could only be established with reference to some compact subset  $\mathcal{D}$  of  $\mathbf{R}^n$ . That is, in practice, one could only get a threshold  $\epsilon^*$  such that, for  $0 < \epsilon \leq \epsilon^*$ ,  $F(x, \epsilon)$  is exact for  $x$  in  $\mathcal{D}$ . In order to measure the degree of exactness of a penalty function, formal definitions of exactness were introduced by Di Pillo and Grippo [13]. By Di Pillo and Grippo's definition [13], for  $0 < 2\alpha\|M\| < 1$ ,  $\Psi_0(x)$  is a strongly exact penalty function for (2.1) with respect to the set  $\mathcal{D} \equiv \mathbf{R}^n$ .

There are two other unconstrained reformulations of (2.1) given by Grippo and Lucidi [13] and by Coleman and Hulbert [5], respectively. Both reformulations are only valid for the case that the feasible region  $\{x : l \leq x \leq u\}$  is compact.

In [13], Grippo and Lucidi started from the definition of differentiable multiplier functions that yield an estimate of the Karush-Kuhn-Tucker multipliers as explicit functions of  $x$ . Then they constructed a penalty function  $P(x, \epsilon)$  containing barrier terms on a perturbation of the constraints. The penalty function  $P(x, \epsilon)$  is a differentiable piecewise rational function in an open neighborhood  $\mathcal{D}$  of the compact feasible region  $\{x : l \leq x \leq u\}$ . The definition of the threshold  $\epsilon^*$  involves the maximum value of  $\|Mx - b\|_\infty$  (the supremum norm of the vector  $(Mx - b)$ ) over the feasible region  $\{x : l \leq x \leq u\}$  and the maximum value of some nonlinear expression of  $x$  and  $\epsilon$ . For  $0 < \epsilon \leq \epsilon^*$ , they proved that  $x^*$  is a global minimizer of  $P(x, \epsilon)$  in  $\mathcal{D}$  if and only if  $x^*$  is a global solution of (2.1). Moreover, any local minimizer  $x^*$  of  $P(x, \epsilon)$  is a local solution of (2.1). The converse is also true when  $(x^*, Mx^* - b)$  is a Karush-Kuhn-Tucker point satisfying strict complementarity conditions. In a separate paper [14], based on the penalty function  $P(x, \epsilon)$ , they proposed Newton-type algorithms to solve (2.1). Under suitable assumptions, finite termination of a Newton method was established.

Grippo and Lucidi's approach was adapted to tackle unbounded feasible sets by Facchinei and Lucidi in [9].

Coleman and Hulbert's penalty function is a nondifferentiable piecewise quadratic function. They started with an  $\ell_1$  penalty term and the transform  $y := -(Mx - b)$  that yield a penalty function  $f(y)$ . Under the assumption that  $M$  is positive definite and the unique solution  $x^*$  of (2.1) is nondegenerate, they proved that a Newton-type method generates a sequence of iterates  $\{y^k\}$  which converge superlinearly to  $y^* := -(Mx^* - b)$  [5]. Note that one has to solve the linear system  $Mx = (b - y^*)$  in order to get the solution  $x^*$ . One interesting feature of Coleman and Hulbert's penalty function is that there is no penalty parameter involved.

In comparison with Grippo and Lucidi's penalty function  $P(x, \epsilon)$  and Coleman and Hulbert's penalty function  $f(y)$ ,  $\Psi_0(x)$  is simpler than  $P(x, \epsilon)$ : a differentiable piecewise quadratic function versus a differentiable piecewise rational function;  $\Psi_0(x)$  is "smoother" than  $P(x, \epsilon)$  or  $f(y)$ : differentiability on  $\mathbf{R}^n$  versus singularity of  $P(x, \epsilon)$  on the boundary of  $\mathcal{D}$  (induced by the barrier terms) or nondifferentiability of  $f(y)$ ;  $\Psi_0(x)$  is "more exact" than  $P(x, \epsilon)$  or  $f(y)$ : equivalence of local solutions without strict complementarity conditions for  $\Psi_0(x)$  versus with strict complementarity conditions for  $P(x, \epsilon)$  or with strict convexity assumption on (2.1) for  $f(y)$ ; the penalty parameter for  $\Psi_0(x)$  is easier to compute than that for  $P(x, \epsilon)$ : the trivial estimate of the threshold versus the complicated estimate of the threshold by maximization of linear and nonlinear functions;  $\Psi_0(x)$  preserves the convexity of the original minimization problem (i.e., if the original quadratic program (2.1) is convex, then  $\Psi_0(x)$  is also convex), while  $P(x, \epsilon)$  or  $f(y)$  does not;  $\Psi_0(x)$  is valid for any feasible set while  $P(x, \epsilon)$  or  $f(y)$  is only defined for compact feasible sets.

The paper is organized as follows. In Section 2, we outline how (2.1) can be reformulated as unconstrained minimization of a quadratic spline penalty function  $\Psi(x)$  and show that the penalty function  $\Psi(x)$  can be derived from Hestenes-Powell-Rockafellar's augmented Lagrangian function by using Fletcher's multiplier function. An important relation between the penalty function  $\Psi(x)$  and the objective function of (2.1) is also given. In Section 3, we relate the second order optimality conditions for (2.1) to those for unconstrained minimization of  $\Psi(x)$ . In Section 4, we prove that (2.1) and the unconstrained reformulation has the same set of local solutions, the same set of isolated local solutions, and the same set of global solutions. Conclusions and final remarks are given in Section 5.

Now we conclude this section by giving some terminologies and notations used in the paper.

For simplicity, we use  $f'(x)$  to denote the gradient of  $f(x)$  (as a column vector) and use  $f''(x)$  to denote the Hessian of  $f(x)$ . A real-valued function  $f(x)$  on  $\mathbf{R}^n$  is said to be a quadratic spline, if the gradient  $f'(x)$  of  $f(x)$  is a piecewise linear mapping from  $\mathbf{R}^n$  to  $\mathbf{R}^n$ . That is, a quadratic spline is a continuously differentiable piecewise quadratic function. The 2-norm  $\|\cdot\|$  on  $\mathbf{R}^n$  is defined as  $\|x\| := (\sum_{i=1}^n x_i^2)^{\frac{1}{2}}$  and the 2-norm of an  $n \times n$  matrix  $B$  is defined as  $\|B\| := \sup\{\|Bx\| : x \in \mathbf{R}^n \text{ with } \|x\| = 1\}$ . The transpose of a matrix  $B$  or a vector  $x$  is denoted by  $B^T$  or  $x^T$ . For  $x, y \in \mathbf{R}^n$ ,  $x \leq y$  means  $x_i \leq y_i$  for  $1 \leq i \leq n$ , where  $x_i$  or  $y_i$  denotes the  $i$ -th component of  $x$  or  $y$ . Let  $(z)_l$  (or  $(z)^u$ ) be the lower (or upper) truncation of  $z$  by  $l$  (or  $u$ ) whose  $i$ -th component is  $\max\{l_i, z_i\}$  (or  $\min\{u_i, z_i\}$ ). By convention,  $z_+$  is a vector whose  $i$ -th component is  $\max\{z_i, 0\}$ . For convenience, we use  $\Lambda$  to denote a diagonal matrix whose diagonal entries are either 0 or 1. The  $n \times n$  identity matrix is written as  $I$  and  $\Lambda_c := I - \Lambda$ . A vector  $x^* \in X$  is said to be a local solution of (2.4) if there exists a positive constant  $\delta$  such that  $f(x) \geq f(x^*)$  for  $x \in X$  with  $\|x - x^*\| \leq \delta$ . A vector  $x^* \in X$  is said to be a isolated local solution of (2.4) if there exists a positive constant  $\delta$  such that  $f(x) > f(x^*)$  for  $x \in X$  and  $0 < \|x - x^*\| \leq \delta$ . A vector  $x^* \in X$  is said to be a global solution of (2.4) if  $f(x) \geq f(x^*)$  for all  $x \in X$ .

## 2.2 Unconstrained Reformulations

In this section we outline how the quadratic program with simple bound constraints, (2.1), can be reformulated as unconstrained minimization of a quadratic spline  $\Psi(x)$  [24, 25], and show that  $\Psi(x)$  can be derived from the Hestenes-Powell-Rockafellar's augmented Lagrangian function by using Fletcher's multiplier function. Finally, an important relation between  $\Psi(x)$  and the objective function of (2.1) is given.

It is well-known from the Karush-Kuhn-Tucker conditions that if  $x$  is a

solution of (2.1), then there exists  $w \in R^n$  such that, for  $1 \leq i \leq n$ ,

$$\begin{aligned} Mx - b - w &= 0, \\ x_i &= l_i \quad \text{if } w_i > 0, \\ x_i &= u_i \quad \text{if } w_i < 0, \\ l_i &\leq x_i \leq u_i \quad \text{if } w_i = 0. \end{aligned} \tag{2.6}$$

One can verify that  $(x^*, w^*)$  is a solution of (2.6) if and only

$$w^* = Mx^* - b \quad \text{and} \quad x^* = (x^* - \alpha w^*)_+^u, \tag{2.7}$$

where  $\alpha$  is any positive constant (cf. [25]). By substituting  $w^* = Mx^* - b$  into (2.7), we observe that (2.6) is equivalent to a system of piecewise linear equations. However, by multiplying a special nonsingular matrix on both sides of (2.7), one can prove that (2.7) is equivalent to the normal equation of a quadratic spline (cf. [25]). The following reformulation of (2.1) follows from Theorem 2.6 in [25].

**Lemma 1** *Suppose that  $M$  is a symmetric matrix. Then  $(x^*, Mx^* - b)$  is a Karush-Kuhn-Tucker point for (2.1) if and only if  $x^*$  satisfies the following piecewise linear equation:*

$$x = (Ex + h)_+^u, \tag{2.8}$$

where  $E := I - \alpha M$ ,  $\alpha$  is any positive constant, and  $h := \alpha b$ . Moreover,  $\frac{1}{\alpha}E(x - (Ex + h)_+^u)$  is the gradient of the following quadratic spline

$$\begin{aligned} \Psi(x) &:= \frac{1}{2\alpha}x^T(E - E^2)x - \frac{1}{\alpha}x^TEh - \frac{\alpha}{2}\|b\|^2 \\ &\quad + \frac{1}{2\alpha}\|(l - (Ex + h))_+\|^2 + \frac{1}{2\alpha}\|((Ex + h) - u)_+\|^2. \end{aligned} \tag{2.9}$$

As a consequence, for  $0 < \alpha\|M\| < 1$ ,  $(x^*, Mx^* - b)$  is a Karush-Kuhn-Tucker point of (2.1) if and only if  $\Psi'(x^*) = 0$ .

**Remark.** Reformulation of (2.6) as a system of piecewise linear equations  $x = (Ex + h)_+^u$  is a generalization of Mangasarian's idea of reformulating a linear complementarity problem as a system of piecewise linear equations. Mangasarian's reformulation led to matrix splitting algorithms for solving the linear complementarity problem [27].

Note that we scale the original  $\Psi_0(x)$  in (2.2) by  $\frac{1}{\alpha}$  and also add an extra term  $-\frac{\alpha}{2}\|b\|^2$ . Lemma 1, not including the last statement, follows directly from Theorem 2.6 in [25]. When  $0 < \alpha\|M\| < 1$ ,  $E := I - \alpha M$  is a positive definite matrix. Therefore,  $\Psi'(x^*) = 0$  if and only if  $\frac{1}{\alpha}E(x^* - (Ex^* + h)_l^u) = 0$ , which is equivalent to  $x^* - (Ex^* + h)_l^u = 0$ . From the first statement of Lemma 1 we know that  $\Psi'(x^*) = 0$  if and only if  $(x^*, Mx^* - b)$  is a Karush-Kuhn-Tucker point of (2.1).

Now let us consider the following unconstrained minimization problem:

$$\min_{x \in \mathbf{R}^n} \Psi(x), \quad (2.10)$$

which is equivalent to (2.2). If  $M$  is positive semidefinite and  $0 < \alpha\|M\| < 1$ , then  $\Psi(x)$  is a convex function. As a consequence,  $x^*$  is a solution of (2.1) if and only if  $x^*$  is a minimizer of  $\Psi(x)$ . Therefore, (2.1) can be reformulated as the unconstrained minimization problem (2.10) when  $M$  is positive semidefinite. However, the reformulation only preserves the first order optimality conditions. When  $M$  is not positive semidefinite, it becomes difficult to relate the second order optimality conditions for (2.1) with the second order optimality conditions for (2.10). Since (2.9) does not clearly show how  $\Psi(x)$  is related to the original objective function  $\frac{1}{2}x^T Mx - b^T x$  and the constraints  $l \leq x \leq u$ , we rewrite  $\Psi(x)$  by substituting  $E = I - \alpha M$  and  $h = \alpha b$  into (2.9). Using simple algebraic manipulations, one can get the following explicit formula for  $\Psi(x)$ .

**Lemma 2** For any  $\alpha > 0$ ,

$$\begin{aligned} \Psi(x) = & \left( \frac{1}{2}x^T Mx - b^T x \right) - \frac{\alpha}{2}\|(Mx - b)\|^2 \\ & + \frac{1}{2\alpha} \left\| ((l - x) + \alpha(Mx - b))_+ \right\|^2 + \frac{1}{2\alpha} \left\| ((x - u) - \alpha(Mx - b))_+ \right\|^2. \end{aligned}$$

**Remark.** The above penalty function can also be derived from Hestenes-Powell-Rockafellar's quadratic augmented Lagrangian function. Consider the following constrained minimization problem:

$$\min_x f(x) \text{ subject to } l \leq g(x) \leq u, \quad (2.11)$$

where  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  and  $g : \mathbf{R}^n \rightarrow \mathbf{R}^m$  are twice continuously differentiable functions. The corresponding augmented Lagrangian function  $L(x, y, \alpha)$  introduced independently by Hestenes [17, 18] and Powell [29] for equality constraints and by Rockafellar [30, 31] for inequality constraints can be written in the following unified way:

$$L(x, y, \alpha) := f(x) + \frac{\alpha}{2} \left\| \left( \frac{1}{\alpha}(g(x) - u) + y \right)_+ \right\|^2 + \frac{\alpha}{2} \left\| \left( \frac{1}{\alpha}(l - g(x)) - y \right)_+ \right\|^2 - \frac{\alpha}{2} \|y\|^2, \quad (2.12)$$

where  $y$  is the Lagrangian multiplier corresponding to two-sided inequality constraints and  $\alpha$  is a penalty parameter. If  $f(x) = \frac{1}{2}x^T Mx - b^T x$  and  $g(x) = x$ , then the corresponding augmented Lagrangian function has the following expression:

$$L(x, y, \alpha) := \frac{1}{2}x^T Mx - b^T x + \frac{\alpha}{2} \left\| \left( \frac{1}{\alpha}(x - u) + y \right)_+ \right\|^2 + \frac{\alpha}{2} \left\| \left( \frac{1}{\alpha}(l - x) - y \right)_+ \right\|^2 - \frac{\alpha}{2} \|y\|^2. \quad (2.13)$$

Note that the Lagrange multiplier  $y$  in (2.13) should satisfy the following equation:

$$y = -(Mx - b). \quad (2.14)$$

As a consequence, we have the following relation between  $\Psi(x)$  and the augmented Lagrangian function  $L(x, y, \alpha)$ :

$$\Psi(x) \equiv L(x, -(Mx - b), \alpha). \quad (2.15)$$

This is actually Fletcher's idea of getting an exact penalty function depending only on  $x$  [10]. For example, suppose the gradients

$$\{g'_1(x), g'_2(x), \dots, g'_m(x)\}$$

of the constraints are linearly independent for any  $x$ . Let  $g'(x)$  be the  $n \times m$  matrix whose  $i$ -th column is the gradient  $g'_i(x)$  of  $g_i(x)$ . Then the Lagrangian multiplier  $y$  in (2.12) should satisfy the following equation:

$$f'(x) + g'(x)y = 0. \quad (2.16)$$



If (2.16) has a solution  $y(x)$ , then  $y(x)$  has the following expression:

$$y(x) = - \left( (g'(x))^T g'(x) \right)^{-1} (g'(x))^T f'(x),$$

which is Fletcher's multiplier estimate. The function  $L(x, y(x), \alpha)$  can be used as an exact penalty function for the two-sided inequality constrained minimization problem [31]. Fletcher [10] initially used  $y(x)$  to eliminate the Lagrangian multiplier in the augmented Lagrangian function for equality constrained minimization problems. Our derivation of the penalty function  $\Psi(x)$  indicates that Fletcher's idea can also be used for inequality constrained minimization problem. It is interesting to notice that, if we can solve the equation (2.16) to get a unique solution  $x(y)$ , then  $L(x(y), y, \alpha)$  is a penalty function depending only on  $y$ . The unconstrained reformulation of strictly convex quadratic program given in [24, 25] can be derived in this way [31].

Now, one may consider that  $\Psi(x)$  is the sum of the objective function

$$\frac{1}{2} x^T M x - b^T x$$

and a quadratic spline penalty term

$$P(x) := -\frac{\alpha}{2} \| (Mx - b) \|^2 + \frac{1}{2\alpha} \left\| ((l - x) + \alpha(Mx - b))_+ \right\|^2 + \frac{1}{2\alpha} \left\| ((x - u) - \alpha(Mx - b))_+ \right\|^2. \quad (2.17)$$

The penalty function has some important properties that relate  $\Psi(x)$  to the objective function of (2.1).

**Lemma 3** For any  $\alpha > 0$ ,

$$\Psi(x) \leq \left( \frac{1}{2} x^T M x - b^T x \right) \quad \text{for } x \in \mathbf{R}^n \text{ with } l \leq x \leq u. \quad (2.18)$$

Moreover, the equality in (2.18) holds if  $(x, Mx - b)$  is a Karush-Kuhn-Tucker point of (2.1).

**Proof.** Let  $l \leq x \leq u$ . Define  $J_1 := \{i : ((l - x) + \alpha(Mx - b))_i > 0\}$ . Then, for  $i \in J_1$ ,

$$(\alpha(Mx - b))_i \geq ((l - x) + \alpha(Mx - b))_i > 0$$

and

$$((x - u) - \alpha(Mx - b))_i = (l - u)_i - ((l - x) + \alpha(Mx - b))_i < 0.$$

Thus,

$$\begin{aligned} \alpha^2(Mx - b)_i^2 &\geq (((l - x) + \alpha(Mx - b))_+)_i^2 \\ &\quad + (((x - u) - \alpha(Mx - b))_+)_i^2. \end{aligned} \quad (2.19)$$

Similarly, for  $i \in J_2 := \{i : ((x - u) - \alpha(Mx - b))_i > 0\}$ , (2.19) holds. The above argument also shows that

$$((l - x) + \alpha(Mx - b))_+^T ((x - u) - \alpha(Mx - b))_+ = 0 \quad (2.20)$$

and  $J_1 \cap J_2 = \emptyset$  (i.e.,  $J_1$  and  $J_2$  are disjoint). Therefore, by (2.19),

$$\begin{aligned} P(x) &= -\frac{\alpha}{2} \sum_{i=1}^n (Mx - b)_i^2 + \frac{1}{2\alpha} \sum_{i \in J_1} (((l - x) + \alpha(Mx - b))_+)_i^2 \\ &\quad + \frac{1}{2\alpha} \sum_{i \in J_2} (((x - u) - \alpha(Mx - b))_+)_i^2 \leq -\frac{\alpha}{2} \sum_{i \notin J_1 \cup J_2} (Mx - b)_i^2 \leq 0. \end{aligned}$$

Thus, for  $l \leq x \leq u$ ,

$$\Psi(x) = \left( \frac{1}{2} x^T Mx - b^T x \right) + P(x) \leq \left( \frac{1}{2} x^T Mx - b^T x \right).$$

Now suppose that  $(x, Mx - b)$  is a Karush-Kuhn-Tucker point of (2.1). It follows from (2.20) that

$$\begin{aligned} \frac{1}{2\alpha} \left\| ((l - x) + \alpha(Mx - b))_+ \right\|^2 &+ \frac{1}{2\alpha} \left\| ((x - u) - \alpha(Mx - b))_+ \right\|^2 \\ &= \frac{1}{2\alpha} \left\| ((l - x) + \alpha(Mx - b))_+ - ((x - u) - \alpha(Mx - b))_+ \right\|^2. \end{aligned} \quad (2.21)$$

One can easily verify that  $z_i^u = z + (l - z)_+ - (z - u)_+$  for any  $z \in \mathbf{R}^n$ . Let  $z := x - \alpha(Mx - b) \equiv Ex + h$ . Then

$$\begin{aligned} &((l - x) + \alpha(Mx - b))_+ - ((x - u) - \alpha(Mx - b))_+ \\ &= (Ex + h)_i^u - (Ex + h) = \alpha(Mx - b), \end{aligned} \quad (2.22)$$

because  $(Ex + h)_i^u - x = 0$  by Lemma 1. By (2.21) and (2.22),  $P(x) = 0$  and the equality holds in (2.18). ■

## 2.3 Second Order Optimality Conditions

All the results in this section are attempts to establish relationships between the second order optimality conditions of (2.1) and (2.10).

From the definition of  $\Psi(x)$  it is easy to verify that, if  $\Psi(\cdot)$  is twice differentiable at  $x$ , then the Hessian  $\Psi''(x)$  has the following expression:

$$\Psi''(x) = \frac{1}{\alpha}(E - E\Lambda E), \quad (2.23)$$

where  $\Lambda := \text{diag}(\lambda_1, \dots, \lambda_n)$  with

$$\lambda_i := \begin{cases} 1, & \text{if } l_i \leq (Ex + h)_i \leq u_i \\ 0, & \text{otherwise.} \end{cases}$$

Even in the case that  $\Psi(\cdot)$  is not differentiable at  $x$ , we still have a similar expression.

**Theorem 4** *Suppose that  $(x^*, Mx^* - b)$  is a Karush-Kuhn-Tucker point of (2.1). Then there exists a positive constant  $\delta$  such that, for any  $x \in \mathbf{R}^n$  with  $\|x - x^*\| \leq \delta$ ,*

$$\Psi(x) - \Psi(x^*) = \frac{1}{2\alpha}(x - x^*)^T(E - E\Lambda E)(x - x^*), \quad (2.24)$$

where  $\Lambda := \text{diag}(\lambda_1, \dots, \lambda_n)$  with

$$\lambda_i := \begin{cases} 1, & \text{if } l_i < (Ex^* + h)_i < u_i \\ 1, & \text{if } l_i = (Ex^* + h)_i \text{ and } (E(x - x^*))_i \geq 0 \\ 1, & \text{if } u_i = (Ex^* + h)_i \text{ and } (E(x - x^*))_i \leq 0 \\ 0, & \text{otherwise.} \end{cases} \quad (2.25)$$

**Proof.** There exists a positive constant  $\delta$  such that

$$\begin{aligned} l_i < (Ex + h)_i < u_i, & \text{ if } l_i < (Ex^* + h)_i < u_i, \\ l_i > (Ex + h)_i, & \text{ if } l_i > (Ex^* + h)_i, \\ u_i < (Ex + h)_i, & \text{ if } u_i < (Ex^* + h)_i, \end{aligned} \quad (2.26)$$

whenever  $\|x - x^*\| \leq \delta$ . Let  $\|x - x^*\| \leq \delta$  and  $\Lambda := \text{diag}(\lambda_1, \dots, \lambda_n)$  with  $\lambda_i$  given in (2.25). Then, for  $x_\theta := \theta x + (1 - \theta)x^*$  with  $0 \leq \theta \leq 1$ ,  $g(\theta) := \Psi(x_\theta)$  is a differentiable function of  $\theta$  and

$$\begin{aligned} g'(\theta) &= (x - x^*)^T \Psi'(x_\theta) \\ &= \frac{1}{\alpha}(x - x^*)^T (Ex_\theta - E(Ex_\theta + h)_i^u) \\ &= \frac{1}{\alpha}(x - x^*)^T (Ex_\theta - E\Lambda(Ex_\theta + h) - E\Lambda_c(Ex^* + h)_i^u), \end{aligned}$$

where the last equality follows from the definition of  $\Lambda$ . Thus, for  $0 \leq \theta \leq 1$ ,  $\Psi(x_\theta)$  is a quadratic polynomial of  $\theta$  and

$$g''(\theta) = \frac{1}{\alpha}(x - x^*)^T (E - E\Lambda E)(x - x^*).$$

By the Taylor expansion and  $\Psi'(x^*) = 0$ , we have

$$\begin{aligned} \Psi(x) - \Psi(x^*) &= g(1) - g(0) = g'(0) + \frac{1}{2}g''(0) \\ &= (x - x^*)^T \Psi'(x^*) + \frac{1}{2\alpha}(x - x^*)^T (E - E\Lambda E)(x - x^*) \\ &= \frac{1}{2\alpha}(x - x^*)^T (E - E\Lambda E)(x - x^*). \quad \blacksquare \end{aligned}$$

Note that the Hessian of the objective function of (2.1) is  $M$ . Therefore, in order to relate the second order optimality conditions of (2.1) to those of (2.10), we have to study how  $(E - E\Lambda E)$  is related to  $M$ .

Recall that we use  $\Lambda$  to denote a diagonal matrix whose diagonal entries are either 0 or 1, and  $\Lambda_c = I - \Lambda$ . Since  $E = I - \alpha M$ , by simple algebraic manipulations, one can verify the following matrix identities.

**Lemma 5** *For any  $\alpha$ ,*

$$E - E\Lambda E = \Lambda_c - \alpha\Lambda_c M \Lambda_c + \alpha(\Lambda M \Lambda - \alpha M \Lambda M) \quad (2.27)$$

and

$$\begin{aligned} E\Lambda M \Lambda E &= \Lambda M \Lambda - \alpha M \Lambda M - \alpha(\Lambda M \Lambda)^2 \\ &\quad + \alpha\Lambda_c M \Lambda M \Lambda_c + \alpha^2(M \Lambda M \Lambda M). \end{aligned} \quad (2.28)$$

In order to derive the second order optimality conditions of (2.10) from those of (2.1), we give a lower bound estimate of  $x^T(E - E\Lambda E)x$ . This estimate will allow us to prove that local solutions (or isolated local solutions) of (2.1) are also local minimizers (or isolated local minimizers) of  $\Psi(x)$ .

**Lemma 6** Suppose that  $0 < 2\alpha\|M\| < 1$ . Then there exist a positive constant  $\beta$  such that

$$x^T(E - E\Lambda E)x \geq \beta(\|\Lambda_c x\|^2 + \|\Lambda M \Lambda x\|^2) + \alpha x^T E \Lambda M \Lambda E x. \quad (2.29)$$

**Proof.** By the definition of the 2-norm of an  $n \times n$  matrix  $Q$ , for any  $y \in \mathbf{R}^n$ , we have

$$\|Qy\| \leq \|Q\|\|y\|. \quad (2.30)$$

By (2.30) and the Cauchy-Schwarz inequality, we get

$$y^T Q y \leq \|y\| \|Qy\| \leq \|Q\| \|y\|^2. \quad (2.31)$$

Let  $\gamma$  be any positive number. Then

$$\begin{aligned} \|\Lambda M x\|^2 &= \|\Lambda M \Lambda x + \Lambda M \Lambda_c x\|^2 \\ &\leq (\|\Lambda M \Lambda x\| + \|\Lambda M \Lambda_c x\|)^2 \\ &\leq (\|\Lambda M \Lambda x\| + \|\Lambda\| \|M \Lambda_c x\|)^2 \\ &\leq (\|\Lambda M \Lambda x\| + \|M\| \|\Lambda_c x\|)^2 \\ &= \|\Lambda M \Lambda x\|^2 + \|M\|^2 \|\Lambda_c x\|^2 + 2\|M\| \|\Lambda_c x\| \|\Lambda M \Lambda x\| \\ &\leq (1 + \gamma^{-1}) \|\Lambda M \Lambda x\|^2 + \|M\|^2 (1 + \gamma) \|\Lambda_c x\|^2, \end{aligned} \quad (2.32)$$

where the first inequality is by the triangle inequality, the second inequality is from (2.30), the third inequality follows from (2.30) and  $\|\Lambda\| \leq 1$ , and the last inequality follows from  $2st \leq \gamma^{-1}s^2 + \gamma t^2$ . By (2.31) and (2.32) we obtain that

$$\begin{aligned} \alpha^3 x^T M \Lambda M \Lambda M x &\leq \alpha^3 \|M\| \|\Lambda M x\|^2 \\ &\leq \alpha^3 \|M\| ((1 + \gamma^{-1}) \|\Lambda M \Lambda x\|^2 + \|M\|^2 (1 + \gamma) \|\Lambda_c x\|^2). \end{aligned} \quad (2.33)$$

Similarly, by (2.31), (2.30), and  $\|\Lambda\| \leq 1$ , we can derive that

$$\alpha x^T \Lambda_c M \Lambda_c x \leq \alpha \|M\| \|\Lambda_c x\|^2 \quad (2.34)$$

and

$$\alpha^2 x^T \Lambda_c M \Lambda M \Lambda_c x \leq \alpha^2 \|M\|^2 \|\Lambda_c x\|^2. \quad (2.35)$$

By (2.27) and (2.28), we obtain that

$$\begin{aligned} x^T(E - E\Lambda E)x - \alpha x^T E\Lambda M\Lambda E x &= x^T \Lambda_c x - \alpha x^T \Lambda_c M \Lambda_c x \\ &+ \alpha^2 x^T (\Lambda M \Lambda)^2 x - \alpha^2 x^T \Lambda_c M \Lambda M \Lambda_c x - \alpha^3 x^T M \Lambda M \Lambda M x. \end{aligned} \quad (2.36)$$

Finally, by (2.36), (2.33), (2.34), and (2.35), we get the following estimate:

$$\begin{aligned} x^T(E - E\Lambda E)x &\geq (1 - \alpha\|M\| - \alpha^2\|M\|^2 - \alpha^3\|M\|^3(1 + \gamma))\|\Lambda_c x\|^2 \\ &+ \alpha^2(1 - \alpha\|M\|(1 + \gamma^{-1}))\|\Lambda M \Lambda x\|^2 + \alpha x^T E\Lambda M\Lambda E x. \end{aligned} \quad (2.37)$$

Let

$$\gamma := \frac{1}{2} \left( \frac{\alpha\|M\|}{1 - \alpha\|M\|} + \frac{1 - 2\alpha\|M\| + \alpha^4\|M\|^4}{\alpha^3\|M\|^3(1 - \alpha\|M\|)} \right)$$

and

$$\beta := \min\{1 - \alpha\|M\| - \alpha^2\|M\|^2 - \alpha^3\|M\|^3(1 + \gamma), \alpha^2(1 - \alpha\|M\|(1 + \gamma^{-1}))\}.$$

Since  $0 < 2\alpha\|M\| < 1$ , one can verify that

$$0 < \frac{\alpha\|M\|}{1 - \alpha\|M\|} < \gamma < \frac{1 - 2\alpha\|M\| + \alpha^4\|M\|^4}{\alpha^3\|M\|^3(1 - \alpha\|M\|)}. \quad (2.38)$$

Since

$$1 - \alpha\|M\| - \alpha^2\|M\|^2 - \alpha^3\|M\|^3(1 + \gamma) = \frac{1 - 2\alpha\|M\| + \alpha^4\|M\|^4}{1 - \alpha\|M\|} - \gamma\alpha^3\|M\|^3,$$

it is easy to verify that (2.38) implies that  $\beta > 0$ . Finally, (2.29) follows from (2.37). ■

## 2.4 Equivalence of The Quadratic Program and Its Unconstrained Reformulation

By using characterizations of local solutions and isolated local solutions of (2.1), we prove that a local solution (or an isolated local solution) of (2.1) is also a local minimizer (or an isolated local minimizer) of  $\Psi(x)$ . By Lemma 3, it is easy to show that the converse also holds. Moreover,  $x^*$  is a global solution of (2.1) if and only if  $x^*$  is a global minimizer of  $\Psi(x)$ .

**Lemma 7** [37] *Let  $(x^*, Mx^* - b)$  be a Karush-Kuhn-Tucker point of (2.1). Then  $x^*$  is a local solution of (2.1) if and only if  $\bar{x}^T M \bar{x} \geq 0$  for  $\bar{x} \in \mathbf{R}^n$  which satisfies the following conditions:*

$$\begin{aligned}\bar{x}_i &= 0, \text{ if } (Ex^* + h)_i < l_i \text{ or } (Ex^* + h)_i > u_i \\ \bar{x}_i &\geq 0, \text{ if } l_i = (Ex^* + h)_i \\ \bar{x}_i &\leq 0, \text{ if } u_i = (Ex^* + h)_i.\end{aligned}$$

**Lemma 8** [39] *Let  $(x^*, Mx^* - b)$  be a Karush-Kuhn-Tucker point of (2.1). Then  $x^*$  is an isolated local solution of (2.1) if and only if  $\bar{x}^T M \bar{x} > 0$  for any nonzero vector  $\bar{x}$  in  $\mathbf{R}^n$  which satisfies the following conditions:*

$$\begin{aligned}\bar{x}_i &= 0, \text{ if } (Ex^* + h)_i < l_i \text{ or } (Ex^* + h)_i > u_i \\ \bar{x}_i &\geq 0, \text{ if } l_i = (Ex^* + h)_i \\ \bar{x}_i &\leq 0, \text{ if } u_i = (Ex^* + h)_i.\end{aligned}$$

**Remark.** The condition given in Lemma 7 is a special case of McCormick's second order necessary condition for a local minima and the condition given in Lemma 8 is a special case of McCormick's second order sufficient condition for a local minima [40].

**Theorem 9** *Suppose that  $0 < 2\alpha\|M\| < 1$ . If  $x^*$  is a local solution (or an isolated local solution) of (2.1), then  $x^*$  is a local minimizer (or an isolated local minimizer) of  $\Psi(x)$ .*

**Proof.** If  $x^*$  is a local solution (or an isolated local solution) of (2.1), then  $(x^*, Mx^* - b)$  satisfies the Karush-Kuhn-Tucker conditions. By Theorem 4, there exist positive constants  $\delta$  and  $\beta$  such that (2.24) and (2.29) hold. Let  $x \in \mathbf{R}^n$  with  $0 < \|x - x^*\| \leq \delta$ .

Note that  $x^*$  is always a local solution of (2.1). (An isolated local solution is a local solution.) By Lemma 7 and the definition of  $\Lambda$  in (2.25), for  $\bar{x} := \Lambda E(x - x^*)$ , we have  $\bar{x}^T M \bar{x} \geq 0$ ; i.e.,  $(x - x^*)^T E \Lambda M \Lambda E(x - x^*) \geq 0$ . By (2.24) and (2.29),

$$\Psi(x) - \Psi(x^*) \geq \frac{\beta}{2\alpha} (\|\Lambda M \Lambda(x - x^*)\|^2 + \|\Lambda_c(x - x^*)\|^2) \geq 0. \quad (2.39)$$

Therefore,  $x^*$  is a local minimizer of  $\Psi(x)$ .

Now assume that  $x^*$  is an isolated local solution of (2.1). We want to show that  $\Psi(x) - \Psi(x^*) > 0$ . If  $\Lambda_c(x - x^*) \neq 0$  or  $\Lambda M \Lambda(x - x^*) \neq 0$ , then it follows from (2.39) that  $\Psi(x) - \Psi(x^*) > 0$ . Otherwise,

$$\Lambda E(x - x^*) = \Lambda(I - \alpha M)(x - x^*) = \Lambda(I - \alpha M)\Lambda(x - x^*) = \Lambda(x - x^*).$$

Since  $x - x^* \neq 0$  and  $\Lambda_c(x - x^*) = 0$ , we get  $\Lambda E(x - x^*) = \Lambda(x - x^*) \neq 0$ . By Lemma 8 and the definition of  $\Lambda$  in (2.25), for  $\bar{x} := \Lambda E(x - x^*)$ , we have  $\bar{x}^T M \bar{x} > 0$ ; i.e.,  $(x - x^*)^T E \Lambda M \Lambda E(x - x^*) > 0$ . Therefore, by (2.24) and (2.29),  $\Psi(x) - \Psi(x^*) > 0$ . Hence,  $x^*$  is an isolated local minimizer of  $\Psi(x)$ . ■

**Theorem 10** *Suppose that  $0 < \alpha \|M\| < 1$ . If  $x^*$  is a local minimizer (or an isolated local minimizer) of  $\Psi(x)$ , then  $x^*$  is a local solution (or an isolated local solution) of (2.1).*

**Proof.** Let  $f(x) := \frac{1}{2}x^T M x - b^T x$ . Since  $x^*$  is a local minimizer of  $\Psi(x)$ ,  $\Psi'(x^*) = 0$  and there exists a positive constant  $\delta$  such that

$$\Psi(x) \geq \Psi(x^*) \quad \text{for } x \in \mathbf{R}^n \text{ with } 0 < \|x - x^*\| \leq \delta. \quad (2.40)$$

By Lemma 1,  $(x^*, Mx^* - b)$  is a Karush-Kuhn-Tucker point of (2.1). Thus, by Lemma 3,  $\Psi(x^*) = f(x^*)$ . It follows from Lemma 3 that

$$f(x) \geq \Psi(x) \geq \Psi(x^*) = f(x^*),$$

whenever  $0 < \|x - x^*\| \leq \delta$  and  $l \leq x \leq u$ . Therefore,  $x^*$  is a local solution of (2.1).

If  $x^*$  is an isolated local minimizer of  $\Psi(x)$ , then the strict inequality holds in (2.40) and, as a consequence,

$$f(x) \geq \Psi(x) > \Psi(x^*) = f(x^*)$$

whenever  $0 < \|x - x^*\| \leq \delta$  and  $l \leq x \leq u$ . Therefore,  $x^*$  is an isolated local solution of (2.1). ■

Finally, we discuss the equivalence of global solutions of (2.1) and global minimizers of  $\Psi(x)$ . First we have the following implication on the existence of global minimizers of  $\Psi(x)$ .



**Theorem 11** Suppose that  $0 < \alpha\|M\| < 1$ . If (2.1) has a global solution, then  $\Psi(x)$  is bounded below on  $\mathbf{R}^n$  and has a global minimizer.

**Proof.** Assume the contrary that  $\Psi(x)$  is not bounded below on  $\mathbf{R}^n$ . Then there exist  $x^k$ 's such that

$$\lim_{k \rightarrow \infty} \Psi(x^k) = -\infty. \quad (2.41)$$

Define

$$\begin{aligned} J_l^k &:= \{i : (Ex^k + h)_i \leq l_i\}, \\ J_u^k &:= \{i : (Ex^k + h)_i \geq u_i\}, \\ J_0^k &:= \{i : l_i < (Ex^k + h)_i < u_i\}. \end{aligned}$$

Since the index set  $\{1, 2, \dots, n\}$  has finitely many different subsets, we can select a subsequence  $\{k_j\}$  such that  $J_l^{k_j} \equiv J_l^{k_1}$ ,  $J_u^{k_j} \equiv J_u^{k_1}$ , and  $J_0^{k_j} \equiv J_0^{k_1}$ . Without loss of generality, we may replace the sequence by the subsequence and assume that there exist three subsets  $J_l$ ,  $J_u$ , and  $J_0$  of  $\{1, 2, \dots, n\}$  such that, for all  $k$ ,

$$\begin{aligned} J_l &= \{i : (Ex^k + h)_i \leq l_i\}, \\ J_u &= \{i : (Ex^k + h)_i \geq u_i\}, \\ J_0 &= \{i : l_i < (Ex^k + h)_i < u_i\}. \end{aligned} \quad (2.42)$$

Let  $\Lambda := \text{diag}(\lambda_1, \dots, \lambda_n)$ , where

$$\lambda_i := \begin{cases} 1, & \text{if } i \in J_0 \\ 0, & \text{otherwise.} \end{cases}$$

Since  $E$  is nonsingular, there exists  $x^0 \in \mathbf{R}^n$  such that

$$\begin{aligned} (Ex^0 + h)_i &= l_i, & \text{if } i \in J_l, \\ (Ex^0 + h)_i &= u_i, & \text{if } i \in J_u, \\ (Ex^0 + h)_i &= (Ex^1 + h)_i, & \text{if } i \in J_0. \end{aligned} \quad (2.43)$$

Similarly as in the proof of Theorem 4,  $\Psi(\theta x^k + (1 - \theta)x^0)$  is a quadratic function of  $\theta$  for  $0 \leq \theta \leq 1$  and

$$\Psi(x^k) - \Psi(x^0) = (x^k - x^0)^T \Psi'(x^0) + \frac{1}{2\alpha} (x^k - x^0)^T (E - E\Lambda E) (x^k - x^0). \quad (2.44)$$

By Lemma 6, there exists a positive constant  $\beta$  such that

$$\begin{aligned} \frac{1}{2\alpha}(x^k - x^0)^T(E - E\Lambda E)(x^k - x^0) &\geq \frac{\beta}{2\alpha}\|\Lambda_c(x^k - x^0)\|^2 \\ &+ \frac{\beta}{2\alpha}\|\Lambda M\Lambda(x^k - x^0)\|^2 + \frac{1}{2}(x^k - x^0)^T E\Lambda M\Lambda E(x^k - x^0). \end{aligned} \quad (2.45)$$

In the remaining part of the proof, we want to show that

$$(x^k - x^0)^T \Psi'(x^0) + \frac{1}{2}(x^k - x^0)^T E\Lambda M\Lambda E(x^k - x^0)$$

is bounded below by a linear function of  $(\Lambda_c(x^k - x^0))$  and  $(\Lambda M\Lambda(x^k - x^0))$ . As a consequence,  $\Psi(x^k)$  is bounded below by a strictly convex quadratic function of  $(\Lambda_c(x^k - x^0))$  and  $(\Lambda M\Lambda(x^k - x^0))$ .

Since  $l \leq (Ex^0 + h) \leq u$  and  $l \leq (Ex^0 + h) + \Lambda E(x^k - x^0) \leq u$ , we have

$$\alpha \Psi'(x^0) = E(x^0 - (Ex^0 + h)_l^u) = E(x^0 - (Ex^0 + h)) = \alpha E(Mx^0 - b) \quad (2.46)$$

and

$$\begin{aligned} -\infty < \kappa &\leq f((Ex^0 + h) + \Lambda E(x^k - x^0)) - f(Ex^0 + h) \\ &= (x^k - x^0)^T E\Lambda(M(Ex^0 + h) - b) \\ &\quad + \frac{1}{2}(x^k - x^0)^T E\Lambda M\Lambda E(x^k - x^0), \end{aligned} \quad (2.47)$$

where  $f(x) := \frac{1}{2}x^T Mx - b^T x$ ,  $\kappa := \inf_{l \leq x \leq u} f(x) - f(Ex^0 + h)$ , and the equality in (2.47) is the Taylor expansion for  $f(x)$ . By (2.46) and

$$M(Ex^0 + h) - b = E(Mx^0 - b),$$

we get

$$\Psi'(x^0) - E\Lambda(M(Ex^0 + h) - b) = (E - E\Lambda E)(Mx^0 - b). \quad (2.48)$$

By simple algebraic manipulations, one can derive from (2.27) that

$$E - E\Lambda E = \Lambda_c(E\Lambda_c - \alpha^2 M\Lambda M) + \alpha \Lambda M\Lambda E. \quad (2.49)$$

It follows from (2.48) and (2.49) that

$$\begin{aligned} (x^k - x^0)^T \Psi'(x^0) - (x^k - x^0)^T E\Lambda(M(Ex^0 + h) - b) \\ = p^T \Lambda_c(x^k - x^0) + q^T \Lambda M\Lambda(x^k - x^0), \end{aligned} \quad (2.50)$$

where  $p := (E\Lambda_c - \alpha^2 M\Lambda M)(Mx^0 - b)$  and  $q := \alpha E(Mx^0 - b)$ . From (2.47) and (2.50) we obtain

$$\begin{aligned} (x^k - x^0)^T \Psi'(x^0) + \frac{1}{2}(x^k - x^0)^T E\Lambda M\Lambda E(x^k - x^0) \\ \geq p^T \Lambda_c(x^k - x^0) + q^T \Lambda M\Lambda(x^k - x^0) + \alpha\kappa. \end{aligned} \quad (2.51)$$

Finally, by (2.44), (2.45), and (2.51), we obtain that

$$\Psi(x^k) - \Psi(x^0) \geq h(\Lambda_c(x - x^0), \Lambda M\Lambda(x - x^0)),$$

where

$$h(y, z) := \frac{\beta}{2\alpha}(\|y\|^2 + \|z\|^2) + p^T y + q^T z + \kappa.$$

Since  $h(y, z)$  is a strictly convex quadratic function of  $(y, z)$ , there exists a constant  $\eta$  such that  $h(y, z) \geq \eta$  for all  $y, z \in \mathbf{R}^n$ . As a consequence,  $\Psi(x^k) \geq \Psi(x^0) + \eta > -\infty$ . The contradiction proves that  $\Psi(x)$  is bounded below on  $\mathbf{R}^n$ .

By Frank-Wolfe Theorem [9, 6], if a piecewise quadratic function is bounded below on  $\mathbf{R}^n$ , it has a global minimizer. Therefore,  $\Psi(x)$  has a global minimizer. ■

**Theorem 12** *Let  $0 < \alpha\|M\| < 1$ . Then  $x^*$  is a global solution of (2.1) if and only if  $x^*$  is a global minimizer of  $\Psi(x)$ .*

**Proof.** Assume that  $x^*$  is a global solution of (2.1). Then  $(x^*, Mx^* - b)$  is a Karush-Kuhn-Tucker point of (2.1) and, by Lemma 3,  $\Psi(x^*) = f(x^*)$ , where  $f(x) := \frac{1}{2}x^T Mx - b^T x$ . By Theorem 11,  $\Psi(x)$  also has a global minimizer  $\hat{x}$ . Since  $\Psi'(\hat{x}) = 0$ , by Lemmas 1 and 3,  $(\hat{x}, M\hat{x} - b)$  is a Karush-Kuhn-Tucker point of (2.1) and

$$\Psi(\hat{x}) = f(\hat{x}) \geq f(x^*) = \Psi(x^*).$$

Thus,  $x^*$  is a global minimizer of  $\Psi(x)$ .

Now suppose that  $x^*$  is a global minimizer of  $\Psi(x)$ . By Lemma 3,  $f(x^*) = \Psi(x^*)$  and

$$f(x) \geq \Psi(x) \geq \Psi(x^*) = f(x^*) \quad \text{for } x \in \mathbf{R}^n \text{ with } l \leq x \leq u.$$

Therefore,  $x^*$  is also a global solution of (2.1). ■

## 2.5 Conclusions

We were able to show that, for  $0 < 2\alpha\|M\| < 1$ , the quadratic program (2.1) is “absolutely” equivalent to the unconstrained minimization of the quadratic spline  $\Psi(x)$  in every conceivable way: the two problems have the same set of local solutions, the same set of isolated local solutions, and the same set of global solutions. Moreover, the exact penalty function  $\Psi(x)$  is convex if (2.1) is a convex quadratic program. Majthay and Mangasarian’s characterizations of local solutions [37] and isolated local solutions [39], respectively, of (2.1) were essential in our analysis.

Since  $M$  is symmetric,  $\|M\|_1 = \|M\|_\infty$  and we have the following upper bound of  $\|M\|$ :

$$\|M\| \leq \sqrt{\|M\|_1 \|M\|_\infty} = \max_{1 \leq i \leq n} \sum_{j=1}^n |m_{ij}|,$$

where  $m_{ij}$  is the  $(i, j)$  entry of  $M$ . See Section 2.2 of [19] for details. Therefore,  $0 < 2\alpha\|M\| < 1$  if

$$0 < \alpha < \left( 2 \cdot \max_{1 \leq i \leq n} \sum_{j=1}^n |m_{ij}| \right)^{-1}.$$

This provides a simple and explicit estimate for threshold of the penalty parameter  $\alpha$ .

From the discussion in Section 2, we realize that the exact penalty function  $\Psi(x)$  could actually be derived from Hestenes-Powell-Rockafellar’s augmented Lagrangian function for two-sided inequality constrained problems by using Fletcher’s multiplier function. The process indicates that one may obtain an exact penalty function by eliminating either the primal variable  $x$  or the dual variable multiplier  $y$  from the augmented Lagrangian function by simply using the equations in the Karush-Kuhn-Tucker conditions for the constrained minimization problem. At least the approach works for the quadratic program with simple bound constraints. Further results for general nonlinear programming problems can be found in [31].

Algorithms based on the exact penalty function  $\Psi(x)$  to solve (2.1) were discussed in [24, 25, 19, 20] when  $M$  is positive semidefinite. It would be interesting to study how the exact penalty function  $\Psi(x)$  can be used to

design new numerical algorithms for solving nonconvex quadratic programs with simple bound constraints.

**Acknowledgement.** The author would like to thank Prof. O. L. Mangasarian for providing references [37, 39] on characterizations of local solutions and isolated local solutions of a quadratic program.

# Bibliography

- [1] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.
- [2] D. P. Bertsekas, Projected Newton methods for optimization methods for optimization problems with simple bounds, *SIAM J. Control Optim.*, 20 (1982), 221-246.
- [3] A. Björck, A direct method for sparse least squares problems with lower and upper bounds, *Numer. Math.*, 54 (1988), 19-32.
- [4] T. F. Coleman and L. A. Hulbert, A direct active set algorithm for large sparse quadratic programs with simple bounds, *Math. Programming*, 45 (1990), 180-201.
- [5] T. F. Coleman and L. A. Hulbert, A globally and superlinearly convergent algorithm for convex quadratic programs with simple bounds, *SIAM J. Optim.*, 3 (1993), 298-321.
- [6] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, Global convergence of a class of trust region algorithms for optimization with simple bounds, *SIAM J. Numer. Anal.*, 25 (1988), 433-460.
- [7] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, Testing a class of methods for solving minimization problems with simple bounds on the variables, *Math. Comput.*, 50 (1988), 399-430.
- [8] R. W. Cottle and M. S. Goheen, A special class of large quadratic programs, in "Nonlinear Programming 3", O. L. Mangasarian, R. R. Meyer, and S. M. Robinson eds., Academic Press, New York, 1978.

- [9] R. W. Cottle, G. H. Golub, and R. S. Sacher, On the solution of large, structured linear complementarity problems: the block partitioned case, *Appl. Math. Optim.*, 4 (1978), 347-363.
- [10] R. W. Cottle, J.-S. Pang, and R. E. Stone, *The Linear Complementarity Problem*, Academic Press, Boston, 1992.
- [11] A. Dax, On computational aspects of bounded linear least squares problems, *ACM Trans. Math. Software*, 17 (1991), 64-73.
- [12] M. A. Diamond, The solution of a quadratic programming problem using fast methods to solve systems of linear equations, *Internat. J. Systems Sci.*, 5 (1974), 131-136.
- [13] G. Di Pillo and L. Grippo, Exact penalty functions in constrained optimization, *SIAM J. Control Optim.*, 27 (1989), 1333-1360.
- [14] F. Facchinei and S. Lucidi, A class of penalty functions for optimization problems with bound constraints, *Optimization*, 26 (1992), 239-259.
- [15] R. Fletcher, A class of methods for nonlinear programming with termination and convergence properties, in "Integer and Nonlinear Programming", Edited by J. Abadie, pp. 157-173. North-Holland Publication, Amsterdam, 1970.
- [16] R. Fletcher and M. P. Jackson, Minimization of a quadratic function of many variables subject to only lower and upper bounds, *J. IMA*, 14 (1974), 159-174.
- [17] M. Frank and P. Wolfe, An algorithm for quadratic programming, *Naval Research Logistics Quarterly*, 3 (1956), 95-110.
- [18] A. Friedlander and J. M. Martínez, On the maximization of a concave quadratic function with box constraints, *SIAM J. Optim.*, 4 (1994), 177-192.
- [19] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The John Hopkins University Press, Baltimore, 1983.

- [20] L. Grippo and S. Lucidi, A differentiable exact penalty function for bound constrained quadratic programming problems, *Optimization*, 22 (1991), 557-578.
- [21] L. Grippo and S. Lucidi, On the solution of a class of quadratic programs using a differentiable exact penalty function, in "System Modelling and Optimization (Leipzig, 1989)", H. J. Sebastian and K. Tammer eds., pp. 764-773. *Lecture Notes in Control and Inform. Sci.*, 143, Springer, Berlin, 1990.
- [22] C.-H. Han, P. M. Pardalos, and Y. Ye, Computational aspects of an interior point algorithm for quadratic programming problems with box constraints, in "Large-scale Numerical Optimization", Edited by Thomas F. Coleman and Yuying Li, pp. 92-112. SIAM, Philadelphia, 1990.
- [23] C.-G. Han, P.M. Pardalos and Y. Ye, An interior point algorithm for large-scale quadratic problems with box constraints, *Springer-Verlag Lecture Notes in Control and Information*, Vol. 144 (Eds: A. Bensoussan and J.L. Lions, 1990), pp. 413-422.
- [24] C.-G. Han, P.M. Pardalos and Y. Ye, On the solution of indefinite quadratic problems using an interior point algorithm, *Informatica*, Vol. 3, No. 4 (1992), pp. 474-496.
- [25] S. P. Han and O. L. Mangasarian, Exact penalty functions in nonlinear programming, *Math. Programming*, 17 (1979), 251-269.
- [26] M. R. Hestenes, Multiplier and gradient methods, in "Computing Methods in Optimization Problems - 2", Edited by L. A. Zadeh, L. W. Neustadt, and A. V. Balakrishnan, pp. 243-164. Academic Press, Newt York, 1969.
- [27] M. R. Hestenes, Multiplier and gradient methods, *J. Optim. Theory Appl.*, 4 (1969), 303-320.
- [28] J. J. Júdice and F. M. Pires, Bard-type methods for the linear complementarity problem with symmetric positive definite matrices, *IMA J. Math. Appl. Business and Industry*, 2 (1989), 51-68.



- [29] W. Li, Linearly convergent descent methods for unconstrained minimization of a convex quadratic spline, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, March 1993, (to appear in J. Optim. Theory Appl.).
- [30] W. Li, A conjugate gradient method for strictly convex quadratic programs with simple bound constraints, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, February 1994.
- [31] W. Li, Differentiable exact penalty functions via Hestenes-Powell-Rockafellar's augmented Lagrangian function, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, April, 1994.
- [32] W. Li and J. Swetits, A Newton method for convex regression, data smoothing, and quadratic programming with bounded constraints, SIAM J. Optim., 3 (1993), 466-488.
- [33] W. Li and J. Swetits, A new algorithm for strictly convex quadratic programs, Technical Report TR92-1, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, Nov. 1992. (to appear in SIAM J. Optim.)
- [34] Y. Lin and C. W. Cryer, An alternating direction implicit algorithm for the solution of linear complementarity problems arising from free boundary problems, Appl. Math. Optim., 13 (1985), 1-17.
- [35] Y. Y. Lin and J.-S. Pang, Iterative methods for large convex quadratic programs: A survey, SIAM J. Control Optim., 25 (1987), 383-411.
- [36] P. Lotstedt, Solving the minimal least squares problem subject to bounds on the variables, BIT, 24 (1984), 206-224.
- [37] A. Majthay, Optimality conditions for quadratic programming, Math. Programming, 1 (1971), 359-365.
- [38] O. L. Mangasarian, Solution of symmetric linear complementarity problems by iterative methods, J. Optim. Theory Appl., 22 (1977), 465-485.

- [39] O. L. Mangasarian, Locally unique solutions of quadratic programs, linear and nonlinear complementarity problems, *Math. Programming*, 19 (1980), 200-212.
- [40] G. P. McCormick, Second order conditions for constrained minima, *SIAM J. Applied Math.*, 15 (1967), 641-652.
- [41] J. J. Moré and G. Toraldo, Algorithms for bound constrained quadratic programming problems, *Numer. Math.*, 55 (1989), 377-400.
- [42] J. J. Moré and G. Toraldo, On the solutions of large quadratic programming problems with bound constraints, *SIAM J. Optim.*, 1 (1991), 93-113.
- [43] D. P. O'Leary, A generalized conjugate gradient algorithm for solving a class of quadratic programming problems, *Linear Alg. Appl.*, 34 (1980), 371-399.
- [44] P. M. Pardalos, C. G. Han and Y. Ye, Interior-point algorithms for solving nonlinear optimization problems, *Math. Progr. Society, COAL Newsletter*, 19 (1991), 45-54.
- [45] P. M. Pardalos, Y. Ye, and C. G. Han, An interior-point algorithm for large-scale quadratic problems with box constraints, in "Analysis and Optimization of Systems", A. Bensoussan and J. L. Lions eds., pp. 413-422. *Lecture Notes in Control and Information Sciences*, 144, Springer-Verlag, NY, 1990.
- [46] R. T. Rockafellar, New applications of duality in convex programming, *Proc. Confer. Probab.*, 4th, Brasov, Romania, 1971, pp. 73-81.
- [47] M. J. D. Powell, A method for nonlinear constraints in minimization problems, in "Optimization", Edited by R. Fletcher, Academic Press, New York, 1972.
- [48] R. T. Rockafellar, The multiplier method of Hestenes and Powell applied to convex programming, *J. Optim. Theory Appl.*, 12 (1973), 555-562.

- [49] R. T. Rockafellar, A duality approach to solving nonlinear programming problems by unconstrained optimization, *Math. Programming*, 5 (1973), 354-573.
- [50] A. Stachurski, Monotone sequences of feasible solutions for quadratic programming problems with  $M$ -matrices and box constraints, in "System Modelling and Optimization (Budapest, 1985)", Edited by A. Prekopa, J. Szelezsan and B. Strazicky, pp. 896-902. *Lecture Notes in Control and Information Sciences*, 84, Springer-Verlag, NY, 1985.
- [51] P. Tseng, D. P. Bertsekas, and J. N. Tsitsiklis, Partially asynchronous, parallel algorithms for network flow and other problems, *SIAM J. Control Optim.*, 28 (1990), 678-710.
- [52] E. K. Yang and J. W. Tolle, A class of methods for solving large, convex quadratic programs subject to box constraints, *Math. Programming*, 51 (1991), 223-228.

## Chapter 3

# A Conjugate Gradient Method for Strictly Convex Quadratic Programs with Simple Bound Constraints

In this paper, we show that an analogue of the classical conjugate gradient method converges linearly when applied to solving the problem of unconstrained minimization of a strictly convex quadratic spline. Since a strictly convex quadratic program with simple bound constraints can be reformulated as unconstrained minimization of a strictly convex quadratic spline, the conjugate gradient method is used to solve the unconstrained reformulation and find the solution of the original quadratic program. In particular, if the solution of the original quadratic program is nondegenerate, then the conjugate gradient method finds the solution in finite iterations.

### 3.1 Introduction

Consider the following convex quadratic programming problem:

$$\begin{aligned} \min \quad & \frac{1}{2}x^T Mx - b^T x \\ \text{subject to } & l \leq Ax \leq u, \end{aligned} \tag{3.1}$$

where  $M$  is an  $n \times n$  symmetric positive semidefinite matrix,  $A$  is an  $m \times n$  matrix,  $x, b \in \mathbf{R}^n$ , and  $l, u$  are vectors of  $m$  components with  $l_i \leq u_i$  (some components of  $l, u$  might be  $\pm\infty$ ). See [19] for a survey on iterative methods for solving (3.1). Recently, Li and Swetits reformulated (3.1) as an unconstrained minimization problem with a convex quadratic spline as the objective function, whenever  $M$  is positive definite or  $A$  is the identity matrix [24, 25]. Here we say that a function  $f(x)$  is a quadratic spline if  $f(x)$  is a differentiable piecewise quadratic function. Therefore, algorithms for unconstrained minimization of a convex quadratic spline can be used to solve convex quadratic programs with simple bound constraints and strictly convex quadratic programs. In [24], a Newton method with line search was proposed to solve (3.1) in finite iterations when  $M$  is positive definite and the rows of  $A$  are linearly independent. Later, a modified version of the Newton method was given in [25] which can also find the solution of (3.1) in finite iterations, under the assumption that  $M$  is positive definite. All these finite algorithms are based on Newton methods for solving the unconstrained reformulation of (3.1) and can start with any initial guess. However, these methods involve solving a linear system in each iteration. They are most efficient for problems where  $M$  is a diagonal matrix and  $A$  has a banded structure or other sparse patterns. For general cases, it seems a good idea to use a "cheap" descent method for a good initial guess and then to use a Newton method for a more accurate numerical solution of (3.1). This leads to a general theory of linearly convergent descent methods for finding a minimizer of a convex quadratic spline which is bounded below on  $\mathbf{R}^n$  [19]. It turns out that it is extremely easy to design a linearly convergent descent method for unconstrained minimization of a convex quadratic spline, even if the set of minimizers of the convex quadratic spline is unbounded [19]. It is clear now that a reformulation of (3.1) as unconstrained minimization of a convex quadratic spline allows one to develop new algorithms for solving (3.1) [23, 24, 25, 19]. Somehow, the problem of unconstrained minimization of a strictly convex quadratic spline is very similar to a linear system with a symmetric positive definite matrix, while the problem of unconstrained minimization of a convex quadratic spline is similar to a linear system with a symmetric positive semidefinite matrix. During a discussion with J.-S. Pang, he suggested that a conjugate gradient method might be developed for unconstrained minimization of a (strictly) convex quadratic spline. In this paper, we show that what Pang suggested can be done.

In past, various conjugate gradient methods were proposed for solving the following unconstrained minimization problem:

$$\inf_{x \in \mathbf{R}^n} f(x), \quad (3.2)$$

where  $f(x)$  is a function bounded below on  $\mathbf{R}^n$ . The major concerns in the design of those conjugate gradient methods were how to avoid the computation of Hessians and how to implement inexact line searches so that the algorithm would be computationally efficient. The standard assumptions are that the objective function is twice continuously differentiable and has bounded level sets. The global convergence of a conjugate gradient method means that

$$\liminf_{k \rightarrow \infty} \|f'(x^k)\| = 0, \quad (3.3)$$

where  $f'(x)$  is the gradient of  $f(x)$  and  $\{x^k\}$  are the iterates generated in the following way:

$$\begin{aligned} p^{k+1} &:= -f'(x^k) + \beta_k p^k, \\ x^{k+1} &:= x^k + \alpha_k p^k. \end{aligned} \quad (3.4)$$

Here  $x^k, p^k$  are vectors in  $\mathbf{R}^n$ ,  $x^0$  is the given initial point,  $p^0 := 0$ , and  $\alpha_k, \beta_k$  are scalars which characterize the underlying conjugate gradient method. See [1, 2, 7, 10, 11, 12, 22, 29] for various rules of selecting  $\alpha_k, \beta_k$  to get a globally convergent conjugate gradient method.

When  $f(x)$  is a convex quadratic spline, it is very easy to evaluate its Hessian (if it exists) and one can also use a linear time algorithm for line search [4, 5, 13, 24]. The difficult issues in the design of conjugate gradient methods for solving (3.1) with a general twice differentiable objective function  $f(x)$  will disappear once  $f(x)$  becomes a convex quadratic spline. However, some new difficult problems will emerge. First of all, a quadratic function has a piecewise linear mapping as its gradient and is not twice differentiable in general. Secondly, the set of minimizers of a convex quadratic spline might be unbounded; therefore, we can not assume bounded level sets. But the simple structure of a convex quadratic spline  $f(x)$  allows us to develop conjugate gradient methods which generate iterates  $\{x^k\}$  such that

$$\text{dist}(x^k, X^*) \leq \gamma \cdot (f_0 + \sqrt{f_0}) \left(1 - \frac{\delta}{(1 + f_0)^2}\right)^k \quad \text{for } k \geq 0, \quad (3.5)$$

where  $f_0 := f(x^0) - \inf_{x \in \mathbf{R}^n} f(x)$ ,  $\delta, \gamma$  are positive constants depending only on  $f$ ,  $X^* := \{x^* \in \mathbf{R}^n : f(x^*) = \inf_{x \in \mathbf{R}^n} f(x)\}$  is the set of all minimizers of  $f(x)$ , and  $\text{dist}(x^k, X^*) := \min\{\|x^k - x^*\| : x^* \in X^*\}$  is the distance from  $x^k$  to  $X^*$ . Note that (3.5) implies that any accumulation point of  $\{x^k\}$  is a minimizer of  $f(x)$  and  $\{x^k\}$  converge linearly to the unique minimizer of  $f(x)$  if  $X^*$  is a singleton. Also we can prove that  $\{\|f'(x^k)\|\}$  converge linearly to zero. However, in general, if the solution set is not a singleton, we do not know whether or not the iterates  $\{x^k\}$  converge.

The proposed convergent conjugate gradient methods may be applied to solving a quadratic programming problem whenever it can be reformulated as unconstrained minimization of a convex quadratic spline. In this paper, we only discuss applications to strictly convex quadratic programs with simple bound constraints:

$$\min_{l \leq x \leq u} \frac{1}{2} x^T M x - b^T x, \quad (3.6)$$

where  $M$  is an  $n \times n$  symmetric positive definite matrix,  $b \in \mathbf{R}^n$  (a vector of  $n$  components), and  $l, u$  are vectors of  $n$  components with  $l < u$ . (Note that, if  $l_i = u_i$ , then  $x_i \equiv l_i$  and one can replace  $x_i$  by  $l_i$  in (3.6) and reformulate the problem so that  $l < u$ .) The article by Moré and Toraldo [20] has an extensive list of references on quadratic programs with simple bound constraints.

In 1969, Polyak [23] proposed a conjugate gradient method for solving (3.6). Since then, there were several articles devoted to discussions on conjugate gradient methods for solving (3.6) [21, 26, 20]. However, these conjugate gradient methods do not fit the framework (3.4). In general, for an iterate  $x^k$  which is not feasible, a special procedure must be used to modify  $x^k$  so that the new  $x^k$  satisfies the simple bound constraints  $l \leq x^k \leq u$ . Since (3.6) can be reformulated as unconstrained minimization of a strictly convex quadratic spline function [25], we can use a natural extension of the classical conjugate gradient method to solve the unconstrained reformulation and find the solution of (3.6). Note that this approach is completely different from the existing conjugate gradient methods for solving (3.6), which are closely related to active set methods. If the quadratic programming problem (3.6) has a nondegenerate solution, then our conjugate gradient method finds its solution in **finite iterations**. As we mentioned before [24, 25], one can use a finite Newton method to solve a strictly convex quadric programming problem. However, if the matrix  $M$  is sparse and the matrix inversion for computing the Newton direction is not desirable, then our conjugate gradient

method provides an appealing alternative for solving (3.6).

The paper is organized as follows. In Section 2, we discuss a class of conjugate gradient methods for unconstrained minimization of a convex quadratic spline function and establish the global error estimate (3.5). Then, we prove that a natural extension of the classical conjugate gradient method finds the unique minimizer  $x^*$  of a strictly convex quadratic spline function  $f(x)$  in finite iterations, if  $f(x)$  is twice differentiable at  $x^*$ . In Section 3, we show that if the solution  $x^*$  to (3.6) is nondegenerate, then the strictly convex quadratic spline function obtained in the unconstrained reformulation of (3.6) is twice differentiable at  $x^*$ . As a consequence, we have a conjugate gradient method for (3.6) which generates a sequence of iterates  $\{x^k\}$  such that  $\{x^k\}$  converge linearly to  $x^*$ . Moreover,  $x^k \equiv x^*$  for  $k$  large enough if  $x^*$  is nondegenerate. Some final conclusions are given in Section 4.

Now we conclude this section by giving some terminologies and notations used in the paper.

For simplicity, we use  $f'(x)$  to denote the gradient of  $f(x)$  (as a column vector) and use  $f''(x)$  to denote the Hessian of  $f(x)$ . A real-valued function  $f(x)$  on  $\mathbf{R}^n$  is said to be a quadratic spline, if the gradient  $f'(x)$  of  $f(x)$  is a piecewise linear mapping from  $\mathbf{R}^n$  to  $\mathbf{R}^n$ . That is, a quadratic spline is a continuously differentiable piecewise quadratic function. Note that  $f''(x)$  might not exist if  $f$  is a quadratic spline. However, we can have a collection of closed convex polyhedral subsets  $\{W_i\}_{i=1}^r$  of  $\mathbf{R}^n$  such that  $f(x)$  is a quadratic function on each  $W_i$  and  $\bigcup_{i=1}^r W_i = \mathbf{R}^n$ . Then  $f''(x)$  exists in the interior of each  $W_i$  but might not exist on the boundary of  $W_i$ . For convenience, we use  $H(x)$  to denote the Hessian of  $f$  restricted on some  $W_i$  containing  $x$ . Note that, if  $x$  is contained in several  $W_i$ 's, then one can arbitrarily choose one of such  $W_i$ 's. The 2-norm  $\|\cdot\|$  on  $\mathbf{R}^n$  is defined as  $\|x\| := (\sum_{i=1}^n x_i^2)^{\frac{1}{2}}$ . The transpose of a matrix  $B$  or a vector  $x$  is denoted by  $B^T$  or  $x^T$ . For  $x, y \in \mathbf{R}^n$ ,  $x \leq y$  means  $x_i \leq y_i$  for  $1 \leq i \leq n$ . We say that iterates  $\{x^k\}$  converge linearly to  $x^*$  if there exist positive constants  $\gamma$  and  $\theta$  such that  $0 \leq \theta < 1$  and  $\|x^k - x^*\| \leq \gamma\theta^k$  for  $k \geq 1$ .



## 3.2 Conjugate Gradient Methods for Convex Quadratic Splines

In this section we propose a class of conjugate gradient methods for unconstrained minimization of a convex quadratic spline function  $f(x)$ :

$$f_{\min} := \inf_{x \in \mathbf{R}^n} f(x). \quad (3.7)$$

Here we assume that  $f(x)$  is bounded below on  $\mathbf{R}^n$  (i.e.  $f_{\min} > -\infty$ ). Then it follows from Frank-Wolfe Theorem [9, 6] that the set of minimizers of  $f(x)$  is not empty. That is,  $X^* := \{x \in \mathbf{R}^n : f(x) = f_{\min}\} \neq \emptyset$ . By using a global error bound for approximate solutions of a convex piecewise quadratic program, we establish global error estimates for iterates generated by the proposed conjugate gradient methods. As a consequence, we prove the linear convergence of an analogue of the classical conjugate method when  $f(x)$  is strictly convex. If, in addition, the strictly convex quadratic spline  $f(x)$  is twice differentiable at its unique minimizer  $x^*$ , then the analogue of the classical conjugate gradient method finds  $x^*$  in finite iterations.

First we formulate a class of conjugate gradient methods based on a sequence of positive definite matrices  $\{D^k\}$ .

**Algorithm 13** For any given  $x^0$  and  $p^0 = 0$  in  $\mathbf{R}^n$ , generate a sequence of iterates as follows:

- (13.1) let  $r^k := -f'(x^k)$ ;
- (13.2) choose a positive definite matrix  $D^k$ ;
- (13.3) set  $\beta_k := -\frac{(r^k)^T D^k p^k}{(p^k)^T D^k p^k}$  if  $p^k \neq 0$  or  $\beta_k := 0$  if  $p^k = 0$ ;
- (13.4) compute the descent direction  $p^{k+1} := r^k + \beta_k p^k$ ;
- (13.5) find the step size  $t_k > 0$  such that  $(p^{k+1})^T f'(x^k + t_k p^{k+1}) = 0$ ;
- (13.6) set  $x^{k+1} := x^k + t_k p^{k+1}$ .

**Remark.** In general, we choose  $D^k := H(x^k)$ , whenever  $H(x^k)$  is positive definite. This is always possible if  $f(x)$  is strictly convex. When  $f''(x^k)$  is only positive semidefinite, we might choose  $D^k = H(x^k) + \epsilon I$  as an approximation of  $H(x^k)$ , where  $\epsilon$  is a fixed positive constant and  $I$  is the identity matrix. In the case that  $f(x)$  is a strictly convex quadratic function, one can easily verify that the above method, with  $D^k = f''(x^k)$ , is the classical conjugate gradient method.

**Theorem 14** Assume that there exists a positive constant  $\epsilon$  such that

$$\epsilon \|x\|^2 \leq x^T D^k x \leq \epsilon^{-1} \|x\|^2$$

for  $x \in \mathbf{R}^n$  and  $k = 0, 1, \dots$ . Let  $\{x^k\}$  be the sequence of iterates generated by Algorithm 13. Then there exist two positive constants  $\delta \equiv \delta(f, \epsilon)$  and  $\gamma \equiv \gamma(f)$  such that

$$\text{dist}(x^k, X^*) \leq \gamma \cdot (f_0 + \sqrt{f_0}) \left(1 - \frac{\delta}{(1 + f_0)^2}\right)^k \quad \text{for } k \geq 0, \quad (3.8)$$

where  $f_0 := f(x^0) - f_{\min}$ .

**Proof.** It is proved in [25] that, if  $f$  is a convex quadratic spline, then there exists a positive constant  $\alpha$  (depending only on  $f$ ) such that

$$\left(\frac{p^T f'(x)}{\|p\|}\right)^2 \leq \alpha (f(x) - f(x + tp)),$$

whenever  $p^T f'(x) < 0$  and  $p^T f'(x + tp) = 0$  (cf. Lemma 3.1 in [25]).

By (13.5) in Algorithm 13, we have

$$(p^{k+1})^T f'(x^k + t_k p^{k+1}) = 0$$

and

$$(p^{k+1})^T f'(x^k) = (r^k)^T f'(x^k) + \beta_k (p^k)^T f'(x^k) = -\|r^k\|^2 < 0. \quad (3.9)$$

Therefore,

$$\left(\frac{(p^{k+1})^T f'(x^k)}{\|p^{k+1}\|}\right)^2 \leq \alpha (f(x^k) - f(x^{k+1})). \quad (3.10)$$

Since  $p^T D^k p \geq \epsilon \|p\|^2$ , we derive that

$$\begin{aligned} \|p^{k+1}\|^2 &\leq \frac{1}{\epsilon} (p^{k+1})^T D^k p^{k+1} \\ &= \frac{1}{\epsilon} ((r^k)^T D^k r^k + 2\beta_k (r^k)^T D^k p^k + \beta_k^2 (p^k)^T D^k p^k) \\ &= \frac{1}{\epsilon} ((r^k)^T D^k r^k - \beta_k^2 (p^k)^T D^k p^k) \\ &\leq \frac{1}{\epsilon} ((r^k)^T D^k r^k), \end{aligned}$$

where the last equality follows from the definition of  $\beta_k$ . Since  $r^T D^k r \leq \frac{1}{\epsilon} \|r\|^2$ , by the above inequality, we obtain

$$\|p^{k+1}\|^2 \leq \frac{1}{\epsilon^2} \|r^k\|^2. \quad (3.11)$$

It follows from (3.9) and (3.11) that

$$\left| (p^{k+1})^T f'(x^k) \right| = |f'(x^k)|^2 \geq \epsilon^2 \|p^{k+1}\|^2. \quad (3.12)$$

By (3.10) and (3.12) we get

$$|f'(x^k)|^2 \leq \frac{\alpha}{\epsilon^2} (f(x^k) - f(x^{k+1})). \quad (3.13)$$

Since  $f(x)$  is a convex quadratic spline, by Theorem 2.2 in [19], (3.13) implies that there exists a positive constant  $\delta$  (depending only on  $f$  and  $\frac{\alpha}{\epsilon^2}$ ) such that

$$f(x^k) - f_{\min} \leq f_0 \left( 1 - \frac{\delta}{(1 + f_0)^2} \right)^k, \quad k = 0, 1, \dots, \quad (3.14)$$

where  $f_0 := f(x^0) - f_{\min} \geq 0$ .

By the global error estimate for a feasible solution of a convex piecewise quadratic program (cf. Corollary 2.8 in [15]), there exists a positive constant  $\gamma$  (depending only on  $f$ ) such that

$$\text{dist}(x^k, X^*) \leq \gamma \left( f(x^k) - f_{\min} + \sqrt{f(x^k) - f_{\min}} \right). \quad (3.15)$$

By (3.14), we have

$$\sqrt{f(x^k) - f_{\min}} \leq \sqrt{f_0} \left( 1 - \frac{\delta}{(1 + f_0)^2} \right)^{\frac{k}{2}} \leq \sqrt{f_0} \left( 1 - \frac{\delta}{(1 + f_0)^2} \right)^k. \quad (3.16)$$

It follows from (3.14), (3.15), and (3.16) that

$$\text{dist}(x^k, X^*) \leq \gamma(f_0 + \sqrt{f_0}) \left( 1 - \frac{\delta}{(1 + f_0)^2} \right)^k.$$

This completes the proof of Theorem 14.

**Remark.** The restriction on  $\{D^k\}$  means that  $\{D^k\}$  is a sequence of uniformly bounded matrices and is also bounded away from the set of singular matrices. It is easy to see that, if  $D^k$ 's are chosen from a finite collection of positive definite matrices, then  $\{D^k\}$  satisfies the assumption made in Theorem 14. In particular,  $D^k := H(x^k)$  satisfies the assumption made in Theorem 14 if  $f(x)$  is strictly convex.

Note that, if  $f(x)$  has a unique minimizer  $x^*$ , then  $\text{dist}(x^k, X^*) = \|x^k - x^*\|$  and the sequence of iterates  $\{x^k\}$  converges linearly to the minimizer  $x^*$  of  $f(x)$ . Theoretically, regardless of the choice of  $D^k$ 's,  $\{x^k\}$  always converges linearly to  $x^*$ . However, a better choice of  $D^k$ 's may increase the value of  $\delta$  and speed up the convergence of  $\{x^k\}$ . The next theorem shows that  $D^k := H(x^k)$  is generally a sound choice for  $D^k$ 's, especially if  $f(x)$  is twice differentiable at  $x^*$ .

**Theorem 15** *Suppose that  $f(x)$  is a strictly convex quadratic spline and  $x^*$  is the unique minimizer of  $f(x)$ . For  $k \geq 1$ , define*

$$D^k := \begin{cases} I, & \text{if } H(x^k) \neq H(x^{k-1}) \\ I, & \text{if } f'(x^{k-1}) \neq f'(x^k) + H(x^k)(x^{k-1} - x^k) \\ H(x^k), & \text{otherwise.} \end{cases}$$

*Then the iterates  $\{x^k\}$  generated by Algorithm 13 converge linearly to  $x^*$ . If, in addition,  $f''(x^*)$  exists and  $H(x) = f''(x)$  whenever  $f''(x)$  exists, then the iterates  $\{x^k\}$  converge finitely to  $x^*$ . That is,  $x^k \equiv x^*$  when  $k$  is large enough.*

**Proof.** Since there are only finitely many distinct  $H(x^k)$ 's,  $D^k$ 's satisfy the assumption of Theorem 14. Thus, the iterates  $\{x^k\}$  converge linearly to the unique minimizer  $x^*$  of  $f(x)$ . Since  $f''(x^*)$  exists, by Taylor expansion, we have

$$f(x) = f(x^*) + (f'(x^*))^T(x - x^*) + \frac{1}{2}(x - x^*)^T f''(x^*)(x - x^*) + o(\|x - x^*\|^2). \quad (3.17)$$

If  $W_i$  contains  $x^*$ , then there exist a vector  $h$  and a matrix  $B$  such that

$$f(x) = f(x^*) + h^T(x - x^*) + \frac{1}{2}(x - x^*)^T B(x - x^*) \quad \text{for } x \in W_i, \quad (3.18)$$

since  $f(x)$  is quadratic on  $W_i$ . For any  $x \in W_i$ , let  $x_t = x^* + t(x - x^*)$ . Then  $x_t \in W_i$  for  $0 \leq t \leq 1$ . It follows from (3.17) and (3.18) that, for  $0 \leq t \leq 1$ ,

$$\begin{aligned} th^T(x - x^*) + \frac{t^2}{2}(x - x^*)^T B(x - x^*) &= f(x_t) \\ &= t(f'(x^*))^T(x - x^*) + \frac{t^2}{2}(x - x^*)^T f''(x^*)(x - x^*) + o(t^2\|x - x^*\|^2). \end{aligned}$$

As a consequence, for any  $x \in W_i$ ,

$$\begin{aligned} h^T(x - x^*) &= (f'(x^*))^T(x - x^*), \\ (x - x^*)^T B(x - x^*) &= (x - x^*)^T f''(x^*)(x - x^*). \end{aligned}$$

The above argument shows that, if  $x^* \in W_i$ , then

$$f(x) = f(x^*) + (f'(x^*))^T(x - x^*) + \frac{1}{2}(x - x^*)^T f''(x^*)(x - x^*) \quad \text{for } x \in W_i. \quad (3.19)$$

Let  $W^* := \bigcup_{x^* \in W_i} W_i$ . Let  $\delta_i := \text{dist}(x^*, W_i)$  and  $\delta := \min\{\delta_i : x^* \notin W_i\}$ . Since  $W_i$  are closed,  $\delta > 0$ . From the definition of  $\delta$  we know that  $x \in W^*$  if  $\|x - x^*\| < \delta$ . Therefore,  $x^*$  is in the interior of  $W^*$ . Moreover, by (3.19),  $f(x)$  is actually a quadratic function on  $W^*$ .

Since  $x^*$  is in the interior of  $W^*$  and  $\{x^k\}$  converge to  $x^*$ , there exists an integer  $k_0 \geq 0$  such that  $x^k$  is in the interior of  $W^*$  for  $k \geq k_0$ . Since  $f(x)$  is a quadratic function on  $W^*$ ,  $f''(x) = f''(x^*)$  for  $x$  in the interior of  $W^*$ . Thus, for  $k > k_0$ ,

$$\begin{aligned} D^k &= H(x^k) = f''(x^k) \equiv f''(x^*), \\ f'(x^{k-1}) &= f'(x^k) + H(x^k)(x^{k-1} - x^k). \end{aligned} \quad (3.20)$$

Let  $k^*$  be the smallest nonnegative integer such that (3.20) holds for  $k > k^*$ . Then, by the definition of  $D^k$ , we get  $D^{k^*} = I$ . Since  $(p^{k^*})^T r^{k^*} = 0$ , we have  $p^{k^*+1} = r^{k^*}$ . Therefore,  $\{x^k\}_{k \geq k^*}$  is a sequence of iterates generated by the classical conjugate gradient method applied to the strictly convex quadratic function

$$\begin{aligned} g(x) &:= f(x^*) + (x - x^*)^T f'(x^*) + \frac{1}{2}(x - x^*)^T f''(x^*)(x - x^*) \\ &\equiv f(x^*) + \frac{1}{2}(x - x^*)^T f''(x^*)(x - x^*). \end{aligned}$$

Thus, we have  $x^k \equiv x^*$  for  $k \geq k^* + n$ .

**Remark.** Note that  $(p^k)^T r^k = 0$  for all  $k$ . Therefore, if  $D^k = I$ , then  $p^{k+1} = r^k$  and the conjugate gradient algorithm automatically restarts at  $x^k$ . The condition for  $D^k = H(x^k)$  is that  $f(x)$  is the same quadratic function at  $x^k$  and  $x^{k-1}$ . Suppose that  $f(x)$  has a unique quadratic representation on each polyhedral region  $W_i$ . Then the choice of  $D^k$  guarantees that the conjugate gradient algorithm restarts whenever the new iterate enters a polyhedral region different from the current one. It seems appropriate to restart in a new region since  $f(x)$  has a new quadratic form there. The only problem is that it degenerates to the steepest descent method if the iterates keep moving from one region to another region. However, the twice differentiability of  $f(x)$  at the solution ensures that the iterates will stay in one region eventually. In general, it is not easy to decide when to restart the conjugate gradient method. Powell [29] did a thorough analysis on numerous restart strategies, including Fletcher and Reeves's restart in  $n$  or  $(n + 1)$  iterations [7] and Beale's restart procedure [3]. Without restarts, Powell [24] shows that the conjugate gradient method usually has a linear rate of convergence.

In application, it is convenient to use  $H(x)$ , the Hessian of  $f(x)$  restricted in some  $W_i$  containing  $x$ , instead of  $f''(x)$  (cf. the next section). If  $f''(x)$  exists and  $W_i$  has nonempty interior, then  $H(x) (\equiv f''(x))$  is independent of the choice of  $W_i$ . However, if  $W_i$  has no interior points, then  $W_i$  is contained in an  $(n - 1)$ -dimensional hyperplane in  $\mathbf{R}^n$ . In this case, there are many different quadratic functions whose restriction on  $W_i$  are the same as the restriction of  $f(x)$  on  $W_i$ . This means that  $H(x)$  might not be the same as  $f''(x)$  even though  $f''(x)$  exists. For example, consider  $f(x_1, x_2) = (x_1 + x_2)_+^2 + \lambda(\lambda - 1 - x_1 - x_2)_+^2$ , where  $\lambda$  is a fixed constant satisfies  $0 < \lambda \leq 1$ . Then it is natural to consider  $f$  as a convex quadratic spline over 4 convex polyhedral regions:

$$\begin{aligned} W_1 &:= \{(x_1, x_2) : (x_1 + x_2) \geq 0, (\lambda - 1 - x_1 - x_2) \geq 0\}, \\ W_2 &:= \{(x_1, x_2) : (x_1 + x_2) \geq 0, (\lambda - 1 - x_1 - x_2) \leq 0\}, \\ W_3 &:= \{(x_1, x_2) : (x_1 + x_2) \leq 0, (\lambda - 1 - x_1 - x_2) \geq 0\}, \\ W_4 &:= \{(x_1, x_2) : (x_1 + x_2) \leq 0, (\lambda - 1 - x_1 - x_2) \leq 0\}. \end{aligned}$$

Obviously, for  $0 < \lambda < 1$ ,  $W_1$  is empty and the Hessian of  $f$  on  $W_4$  is  $\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$ .

However, if  $\lambda = 1$ , then  $f(x_1, x_2) \equiv (x_1 + x_2)^2$  and  $f''(x) \equiv \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$ . In this case,  $W_1 = W_4$  and all the regions contain the origin. However, if we use  $H(0,0) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$  as the Hessian of  $f$  on  $W_4$  (which degenerates to a point), then  $H(0,0) \neq f''(0,0)$ . When a polyhedral region  $W_i$  is defined by linear inequalities  $Ax \geq b$ ,  $W_i$  has nonempty interior if and only if  $Ax \geq b$  satisfies the Slater condition (i.e.,  $A\hat{x} > b$  for some  $\hat{x}$ ). There is no simple way to check whether a polyhedral region has interior point or not. Thus, we can not expect that our choice of  $W_i$  guarantee  $H(x) = f''(x)$  when  $f$  is twice differentiable at  $x$ . That is the reason why we have to assume that  $H(x) = f''(x)$  in the above theorem. However, if all  $W_i$  have nonempty interior, then  $H(x) \equiv f''(x)$  whenever  $f''(x)$  exists (cf. the proof of Theorem 17).

### 3.3 Strictly Convex Quadratic Programs with Simple Bound Constraints

In this section we outline how the strictly convex quadratic program with simple bound constraints, (3.6), can be reformulated as unconstrained minimization of a strictly convex quadratic spline  $\Psi(x)$  [24, 25]. As a consequence, we have an analogue of Theorem 15 for (3.6). An very interesting result is that  $\Psi(x)$  is twice differentiable if (3.6) has a nondegenerate solution (cf. the proof of Theorem 17).

It is well-known from the Karush-Kuhn-Tucker conditions that  $x$  is a solution of (3.6) if and only if there exists  $w \in R^n$  such that, for  $1 \leq i \leq n$ ,

$$\begin{aligned} Mx - b - w &= 0, \\ x_i &= l_i \quad \text{if } w_i > 0, \\ x_i &= u_i \quad \text{if } w_i < 0, \\ l_i &\leq x_i \leq u_i \quad \text{if } w_i = 0. \end{aligned} \tag{3.21}$$

Based on the above Karush-Kuhn-Tucker conditions, one can verify that  $x^*$  is a solution of (3.6) if and only

$$x^* = (x^* - \alpha w^*)_i^u, \tag{3.22}$$

where  $w^* = Mx^* - b$  and  $\alpha$  is any positive constant (cf. [25]). Here  $(z)_l$  (or  $(z)_u$ ) is the lower (or upper) truncation of  $z$  by  $l$  (or  $u$ ) whose  $i$ -th component is  $\max\{l_i, z_i\}$  (or  $\min\{u_i, z_i\}$ ). By substituting  $w^* = Mx^* - b$  into (3.22), we observe that (3.6) is equivalent to a system of piecewise linear equations. However, by multiplying a special nonsingular matrix on both sides of (3.22), one can prove that (3.22) is equivalent to the normal equation of a convex quadratic spline (cf. [25]). The following lemma follows from Theorem 2.6 in [25].

**Lemma 16** *Suppose that  $M$  is symmetric positive definite. Then  $x^*$  is a solution of (3.6) if and only if  $x^*$  satisfies the following piecewise linear equation:*

$$x = (Ex + h)_l^u,$$

where  $E := I - \alpha M$ ,  $\alpha$  is any positive constant with  $\alpha \|M\| < 1$ ,  $\|M\|$  denotes the spectral radius of  $M$ , and  $h := \alpha b$ . Moreover,  $E(x - (Ex + h)_l^u)$  is the gradient of the following strictly convex quadratic spline

$$\Psi(x) := \frac{1}{2}x^T(E - E^2)x - x^TEh + \frac{1}{2}\|(l - (Ex + h))_+\|^2 + \frac{1}{2}\|((Ex + h) - u)_+\|^2.$$

As a consequence,  $x^*$  is a solution of (3.6) if and only if  $x^*$  is the unique minimizer of  $\Psi(x)$ .

Before applying Algorithm 13 to  $f(x) := \Psi(x)$ , we give an explicit formula for  $H(x)$ . For any  $x \in \mathbf{R}^n$ , let  $\sigma(x)$  be the diagonal matrix whose  $i$ -th diagonal element  $\sigma_{ii}(x)$  is defined by the following formula:

$$\sigma_{ii}(x) := \begin{cases} 1, & \text{if } l_i < (Ex + h)_i < u_i \\ 0, & \text{if } (Ex + h)_i \leq l_i \text{ or } (Ex + h)_i \geq u_i. \end{cases} \quad (3.23)$$

Then one can verify that, if  $\Psi''(x)$  exists, then

$$\Psi''(x) = E - E\sigma(x)E.$$

Thus, we define

$$H(x) := E - E\sigma(x)E. \quad (3.24)$$



One may consider  $H(x^k)$  as the Hessian of  $\Psi(x)$  restricted on the closed convex polyhedral set

$$W(x^k) := \{x \in \mathbf{R}^n : (Ex + h)_i \geq u_i \text{ for } i \in I_u(x^k), (Ex + h)_i \leq l_i \text{ for } i \in I_l(x^k), \text{ and } l_i \leq (Ex + h)_i \leq u_i \text{ for } i \in I_0(x^k)\}, \quad (3.25)$$

where

$$\begin{aligned} I_u(x^k) &:= \{i : (Ex^k + h)_i \geq u_i\}, \\ I_l(x^k) &:= \{i : (Ex^k + h)_i \leq l_i\}, \\ I_0(x^k) &:= \{i : l_i < (Ex^k + h)_i < u_i\}. \end{aligned} \quad (3.26)$$

With  $D^k := H(x^k)$  and  $f'(x) := Ex - E(Ex + h)^u$  in Algorithm 13, we obtain a linearly convergent conjugate gradient algorithm for solving the strictly convex quadratic program with simple bound constraints. Moreover, the nondegeneracy assumption of (3.6) implies the finite termination of the conjugate gradient method. Recall that a solution  $x^*$  of (3.6) is nondegenerate if and only if

$$\begin{aligned} x_i^* &= l_i \quad \text{if } (Mx^* - b)_i > 0, \\ x_i^* &= u_i \quad \text{if } (Mx^* - b)_i < 0, \\ l_i &< x_i^* < u_i \quad \text{if } (Mx^* - b)_i = 0. \end{aligned} \quad (3.27)$$

**Theorem 17** *Suppose that  $M$  is a positive definite matrix. For any vector  $x^0$  in  $\mathbf{R}^n$  and  $p^0 := 0$ , generate a sequence of iterates  $\{x^k\}$ ,  $k = 1, 2, \dots$ , as follows:*

(17.1) *let  $r^k := E(Ex^k + h)^u - Ex^k$ ;*

(17.2) *if  $W(x^k) \neq W(x^{k-1})$  or  $p^k = 0$ , set  $\beta_k := 0$ ; otherwise, set*

$$\beta_k := -\frac{(r^k)^T Ep^k - (Er^k)^T \sigma(x^k)(Ep^k)}{(p^k)^T Ep^k - (Ep^k)^T \sigma(x^k)(Ep^k)};$$

(17.3) *compute the descent direction  $p^{k+1} := r^k + \beta_k p^k$ ;*

(17.4) *find  $t_k > 0$  such that*

$$(Ep^{k+1})^T (x^k + t_k p^{k+1} - (Ex^k + h + t_k Ep^{k+1})^u) = 0;$$

(17.5) *set  $x^{k+1} := x^k + t_k p^{k+1}$ .*

*Then  $\{x^k\}$  converge linearly to the unique solution  $x^*$  of (3.6). Moreover, if  $x^*$  is a nondegenerate solution of (3.6), then  $x^k \equiv x^*$  when  $k$  is large enough.*

**Proof.** Note that, if  $M$  is positive definite, then  $E$  is a positive definite matrix with eigenvalues in the interval  $(0, 1)$ . Thus,  $E - E^2$  is a positive definite matrix. As a consequence,  $x^T(E - E\sigma(x^k)E)x \geq x^T(E - E^2)x > 0$  for  $x \neq 0$ . Therefore,  $D^k := H(x^k)$  is positive definite (cf. [25]). Note that  $\Psi(x)$  is the same quadratic function on  $W(x^k)$  and  $W(x^{k-1})$  if and only if  $W(x^k) = W(x^{k-1})$ . Therefore, one can verify that the steps (17.1)–(17.5) generate the same iterate as Algorithm 13 with  $D^k$  defined as in Theorem 15 and  $f'(x) = Ex - E(Ex + h)_i^u$ . Hence, by Theorem 14,  $\{x^k\}$  converge linearly to  $x^*$ . Note that, if  $f''(x^k)$  exists, then, by the proof of Theorem 15,

$$(x - x^k)^T H(x^k)(x - x^k) = (x - x^k)^T f''(x^k)(x - x^k) \quad \text{for } x \in W(x^k). \quad (3.28)$$

Since  $E$  is positive definite,  $W(x^k)$  has nonempty interior and  $\{x - x^k : x \in W(x^k)\}$  spans  $\mathbf{R}^n$ . Therefore, (3.28) implies  $H(x^k) = f''(x^k)$ . Moreover,  $H(x^k) = H(x^{k-1})$  and To complete the proof, we only need to show that  $\Psi'(x) = Ex - E(Ex + h)_i^u$  is differentiable at  $x^*$  if  $x^*$  is nondegenerate, since Theorem 15 implies the finite convergence of  $\{x^k\}$ .

Let  $x^*$  be a nondegenerate solution of (3.6). Then (cf. (3.25))

$$\begin{aligned} \text{int}W(x^*) &= \{x : (Ex + h)_i < l_i \text{ for } i \in I_l(x^*), (Ex + h)_i < l_i \\ &\quad \text{for } i \in I_l(x^*), \text{ and } l_i < (Ex + h)_i < u_i \text{ for } i \in I_0(x^*)\}. \end{aligned}$$

By (3.27), for any index  $i$ ,  $(Ex^* + h)_i = (x^* - \alpha(Mx^* - b))_i$  can not equal to  $l_i$  or  $u_i$ . Therefore,  $x^* \in \text{int}W(x^*)$ . Since  $\Psi'(x)$  is a linear mapping on the open set  $\text{int}W(x^*)$ ,  $\Psi''(x)$  exists for  $x \in \text{int}W(x^*)$ . In particular,  $\Psi''(x^*)$  exists. This completes the proof of Theorem 17.

**Remark.** We only need to keep a sign pattern vector  $S(x)$  for each polyhedral region  $W(x)$ :  $S_i(x) = 1$  if  $(Ex + h)_i \geq u_i$ ,  $S_i(x) = 0$  if  $l_i < (Ex + h)_i < u_i$ , and  $S_i(x) = -1$  if  $(Ex + h)_i \leq l_i$ . With the storage of  $S(x^{k-1})$  and  $S(x^k)$ , we can easily check whether or not  $W(x^k) = W(x^{k-1})$ , since  $W(x^k) = W(x^{k-1})$  if and only if  $S(x^{k-1}) = S(x^k)$ . Also we can use a linear time algorithm to find the step size  $t_k$  (cf. [4, 5, 13, 24]). Therefore, the most expensive operations in each iteration is the matrix-vector multiplications. Note that there are only four matrix-vector multiplications involved in each iteration:

$$Ex^k, E(Ex^k + h)_i^u, Ep^k, Er^k,$$

since  $Ep^{k+1} = Er^k + \beta_k Ep^k$ . Therefore, roughly speaking, the computational cost of each iteration is about four times of that of the classical conjugate

gradient method for solving the linear system  $Mx - b = 0$ . Also note that the reformulation does not change any sparse pattern of  $M$ .

Finally, we want to say a few words about  $H(x^k)$  defined by (3.24). We can also define  $\sigma(x)$  in the following way:

$$\sigma_{ii}(x) := \begin{cases} 1, & \text{if } l_i \leq (Ex + h)_i \leq u_i \\ 0, & \text{if } (Ex + h)_i < l_i \text{ or } (Ex + h)_i > u_i. \end{cases} \quad (3.29)$$

Then Theorem 17 still holds. However, the choices of  $\sigma(x)$  are related to how one wants to treat the current expected active constraints. See [25] for details.

### 3.4 Conclusions

Based on a conjugate gradient method for unconstrained minimization of a strictly convex quadratic function, we derived a conjugate gradient method for strictly convex quadratic programs with simple bound constraints. We also established a global error estimate for iterates which implies the linear convergence of the iterates. In the case that the solution of the original quadratic program is nondegenerate, then the conjugate gradient method terminates in finite iterations. The conjugate gradient method can start at any point which may be infeasible. The computational cost for each iteration is  $\mathcal{O}(n^2)$  flops, proportional to the classical conjugate gradient method. The study indicates that the strictly convex quadratic program with simple bound constraints (3.6) is very similar to the linear system  $Mx = b$ . Extensive numerical experiments shall be done to compare the proposed method with other existing methods for solving (3.6). The purpose of this article is to show the importance of the reformulation of (3.6) as unconstrained minimization of a strictly convex quadratic spline and to establish a foundation for further study on applications of such an unconstrained reformulation.

Note that the conjugate gradient methods discussed in [1, 2, 7, 10, 11, 12, 22, 29] can be directly applied to an unconstrained minimization problem with a strictly convex quadratic spline as the objective function. It is interesting to know what convergence result one may derive for this special case.

**Acknowledgement.** The author is grateful to Prof. J.-S. Pang for proposing the idea of using conjugate gradient methods for unconstrained minimization of a convex quadratic spline and for many comments later which clarified some proofs in the earlier drafts of this paper. The author would also like to thank Prof. R. Fletcher for kindly pointing out that the original version of the algorithm in Theorem 15, which had no restart feature, could not terminate in finite iterations.

# Bibliography

- [1] D. Touati-Ahmed and C. Storey, Efficient Hybrid conjugate gradient techniques, J. Optim. Theory Appl., 64 (1990), 379-397.
- [2] M. Al-Baali, Descent property and global convergence of the Fletcher-Reeves method with inexact line-search, IMA J. Numer. Anal., 5 (1985), 121-124.
- [3] E. M. L. Beale, A derivation of conjugate gradients, in "Numerical Methods for Nonlinear Optimization", Edited by F. A. Lootsma, pp. 39-43. Academic Press, London, 1972.
- [4] P. Brucker, An  $\mathcal{O}(n)$  algorithm for the quadratic knapsack problem, Oper. Res. Letters, 3 (1984), 163-166.
- [5] P. H. Calamai and J. J. Moré, Quasi-Newton updates with bounds, SIAM J. Numer. Anal., 24 (1987), 1434-1441.
- [6] R. W. Cottle, J.-S. Pang, and R. E. Stone, *The Linear Complementarity Problem*, Academic Press, Boston, 1992.
- [7] R. Fletcher and C. M. Reeves, Function minimization by conjugate gradients, Computer J., 7 (1964), 143-154.
- [8] M. A. Diamond, The solution of a quadratic programming problem using fast methods to solve systems of linear equations, Internat. J. Systems Sci., 5 (1974), 131-136.
- [9] M. Frank and P. Wolfe, An algorithm for quadratic programming, Naval Research Logistics Quarterly, 3 (1956), 95-110.

- [10] Y. F. Hu and C. Storey, Efficient generalized conjugate gradient algorithms, Part 1: Theory, J. Optim. Theory Appl., 69 (1991), 129-137.
- [11] Y. F. Hu and C. Storey, Efficient generalized conjugate gradient algorithms, Part 2: Implementation, J. Optim. Theory Appl., 69 (1991), 139-152.
- [12] Y. F. Hu and C. Storey, Global convergence result for conjugate gradient methods, J. Optim. Theory Appl., 71 (1991), 399-405.
- [13] N. Kover and P. M. Pardalos, An algorithm for a singly constrained class of quadratic programs subject to upper and lower bounds, Math. Programming, 46 (1990), 321-328.
- [14] W. Li, P. Pardalos, and C. G. Han, Gauss-Seidel method for least distance problems, J. Optim. Theory Appl. 75 (1992), 487-500.
- [15] Wu Li, Error bounds for piecewise convex quadratic programs and applications, Technical Report TR93-1, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, Jan., 1993, (to appear in SIAM J. Control Optim.).
- [16] Wu Li, Linearly convergent descent methods for unconstrained minimization of a convex quadratic spline, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, March 1993, (to appear in J. Optim. Theory Appl.).
- [17] W. Li and J. Swetits, A Newton method for convex regression, data smoothing, and quadratic programming with bounded constraints, SIAM J. Optim., 3 (1993), 466-488.
- [18] W. Li and J. Swetits, A new algorithm for strictly convex quadratic programs, Technical Report TR92-1, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, Nov. 1992. (to appear in SIAM J. Optim.)
- [19] Y. Y. Lin and J.-S. Pang, Iterative methods for large convex quadratic programs: A survey, SIAM J. Control Optim., 25 (1987), 383-411.

- [20] J. J. Moré and G. Toraldo, On the solution of large quadratic programming problems with bound constraints, *SIAM J. Optim.*, 1 (1991), 93-113.
- [21] D. P. O'Leary, A generalized conjugate gradient algorithm for solving a class of quadratic programming problems, *Linear Alg. Appl.*, 34 (1980), 371-399.
- [22] E. Polak and G. Ribière, Nore sur la Convergence des Méthodes de directions Conjuguées, *Revue Francaise, Information et Recherche Opérationnelle*, 6 (1969), 35-43.
- [23] B. T. Polyak, The conjugate gradient method in extremal problems, *USSR Comput. Math. and Math. Phys.*, 9 (1969), 94-112.
- [24] M. J. D. Powell, Some convergence properties of the conjugate gradient method, *Math. Programming*, 11 (1976), 42-49.
- [25] M. J. D. Powell, Restart procedures for the conjugate gradient method, *Math. Prog.*, 12 (1977), 241-254.
- [26] E. K. Yang and J. W. Tolle, A class of methods for solving large convex quadratic programs subject to box constraints, *Math. Prog.*, 51 (1991), 223-228.

## Chapter 4

# Differentiable Exact Penalty Functions via Hestenes-Powell-Rockafellar's Augmented Lagrangian Function

We study differentiable exact penalty functions, depending only on  $x$ , derived from Hestenes-Powell-Rockafellar's quadratic augmented Lagrangian function for a minimization problem with two-sided inequality constraints by using Fletcher's Lagrangian multiplier estimate. We also consider new penalty functions, depending only on the Lagrangian multiplier, derived from the augmented Lagrangian function. These penalty functions are particularly useful for quadratic programming problems.

### 4.1 Introduction

Consider the following constrained minimization problem:

$$\min_x f(x) \text{ subject to } l \leq g(x) \leq u, \quad (4.1)$$

where  $f : \mathbf{R}^n \rightarrow \mathbf{R}$  and  $g : \mathbf{R}^n \rightarrow \mathbf{R}^m$  are twice continuously differentiable functions. Here we assume  $l \leq u$  and some components of  $l$  and  $u$  may be



$-\infty$  and  $\infty$ , respectively. The corresponding augmented Lagrangian function  $L(x, y, \alpha)$  introduced independently by Hestenes [17, 18] and Powell [29] for equality constraints and by Rockafellar [30, 31] for inequality constraints can be written in the following unified way:

$$L(x, y, \alpha) := f(x) + \frac{\alpha}{2} \left\| \left( \frac{1}{\alpha}(g(x) - u) + y \right)_+ \right\|^2 + \frac{\alpha}{2} \left\| \left( \frac{1}{\alpha}(l - g(x)) - y \right)_+ \right\|^2 - \frac{\alpha}{2} \|y\|^2, \quad (4.2)$$

where  $y$  is the Lagrangian multiplier corresponding to two-sided inequality constraints,  $\alpha$  is a penalty parameter, and  $(z)_+$  is the vector whose  $i$ th component is  $\max\{0, z_i\}$ . The idea of using one Lagrangian multiplier for two-sided inequality constraints was first proposed by Bertsekas [3, 1, 2]. Along with complementarity conditions, the Lagrangian multiplier  $y$  satisfies the following equation:

$$f'(x) + g'(x)y = 0, \quad (4.3)$$

where  $f'(x)$  is the gradient of  $f(x)$  whose  $i$ th component is  $\frac{\partial f}{\partial x_i}$  and  $g'(x)$  is the Jacobian of  $g(x)$  whose  $i$ th column is the gradient of  $g_i(x)$ . For equality constrained minimization problems ( $l = u = 0$ ), the corresponding augmented Lagrangian function is reduced to the following form:

$$L(x, y, \alpha) = f(x) + \frac{\alpha}{2} \left\| \left( \frac{1}{\alpha}g(x) + y \right)_+ \right\|^2 + \frac{\alpha}{2} \left\| \left( -\frac{1}{\alpha}g(x) - y \right)_+ \right\|^2 - \frac{\alpha}{2} \|y\|^2,$$

i.e.,

$$L(x, y, \alpha) = f(x) + y^T g(x) + \frac{1}{2\alpha} \|g(x)\|^2, \quad (4.4)$$

since

$$\left\| \left( \frac{1}{\alpha}g(x) + y \right)_+ \right\|^2 + \left\| \left( -\frac{1}{\alpha}g(x) - y \right)_+ \right\|^2 = \left\| \frac{1}{\alpha}g(x) + y \right\|^2.$$

For equality constraints, it is natural to assume that  $g'(x)$  has rank  $m$  for every  $x$  and Fletcher proposed to use the following estimate for the multiplier  $y$ :

$$y(x) := -(g'(x)^T g'(x))^{-1} g'(x)^T f'(x). \quad (4.5)$$

Substituting  $y$  in (4.4) by  $y(x)$  given in (4.5), Fletcher obtained a penalty function  $L(x, y(x), \alpha)$  depending only on  $x$  [10]. Glad and Polak [12] considered that Fletcher's idea was very good, except for two shortcomings.

"The first was that he did not know how to find automatically a satisfactory value of the penalty  $\alpha$ , while the other was that his extension of his formula to problems with inequalities [11] results in discontinuous derivatives in the augmented Lagrangian, which caused algorithms to jam." Therefore, for mixed equality and inequality constraints, say  $l_i = u_i = 0$  for  $1 \leq i \leq k$  and  $l_i = -\infty, u_i = 0$  for  $k+1 \leq i \leq m$ , Glad and Polak proposed a new formula for the multiplier  $y$ . Glad and Polak's multiplier function was designed to minimize the violation of Karush-Kuhn-Tucker conditions [12]:

$$\min_y \|f'(x) + g'(x)y\|^2 + \gamma^2 \sum_{i=k+1}^m (g_i(x)y_i)^2, \quad (4.6)$$

where  $\gamma$  is a nonzero scalar. Under the linear independence constraint qualification, (4.6) is a strictly convex quadratic program and its solution  $y^*(x)$  can be explicitly written as follows:

$$y^*(x) = -(g'(x)^T g'(x) + \gamma^2 G(x)^2)^{-1} g'(x)^T f'(x), \quad (4.7)$$

where  $G(x)$  is the diagonal matrix whose  $i$ th diagonal entry is 0 for  $1 \leq i \leq k$  and  $g_i(x)$  for  $k+1 \leq i \leq m$ .

Comparing Fletcher's multiplier  $y(x)$  with Glad and Polak's multiplier  $y^*(x)$ , we realize that  $y(x)$  is actually equal to  $y^*(x)$  with  $\gamma = 0$ , when  $g'(x)$  has rank  $m$ . The matrix  $\gamma^2 G(x)^2$  is somehow a regularization factor and, as a consequence,  $y^*(x)$  is differentiable under the linear independence constraint qualification while  $y(x)$  is only differentiable under a much stronger condition that the gradients  $g'_1(x), \dots, g'_m(x)$  are linearly independent. Therefore, Glad and Polak's multiplier  $y^*(x)$  can be applied to more general cases than Fletcher's multiplier  $y(x)$ . However, within the unified framework for equality and inequality constraints, the restriction of Fletcher's multiplier  $y(x)$  is due to its stronger requirement on the constraint qualification instead of the types of constraints. Even for two-sided inequality constraints  $l \leq g(x) \leq u$ , there is no reason not to use Fletcher's multiplier  $y(x)$  if  $g'(x)$  has rank  $m$ . In fact, in this case,  $y(x)$  is much simpler than  $y^*(x)$  and results in a much better penalty function  $L(x, y(x), \alpha)$ . For example, for linear inequality constraints such as simple bound constraints on the variable  $x$  and a quadratic objective function  $f(x)$ ,  $y^*(x)$  is a rational function of  $x$  while  $y(x)$  is a linear function of  $x$ . As a consequence, it becomes extremely difficult to find a penalty parameter  $\alpha$  that makes  $L(x, y^*(x), \alpha)$  an exact penalty function. In general,

one can only define the penalty parameter  $\alpha$  to get an exact penalty function  $L(x, y^*(x), \alpha)$  with respect to a compact set [7]. Lucidi and Grippo had to incorporate barrier terms into the augmented Lagrangian function to get a global exact penalty function for quadratic programs with simple bound constraints [13, 14]. On the contrary, for quadratic programs with simple bound constraints, the differentiable penalty function  $L(x, y(x), \alpha)$  is a differentiable piecewise quadratic function which is a strongly exact penalty function with respect to  $\mathbf{R}^n$  (a noncompact set) and the penalty parameter  $\alpha$  can be easily estimated by using the spectrum radius of the Hessian of the quadratic objective function [21] (cf. also Proposition 29). In fact, for quadratic programs with linearly independent two-sided inequality constraints, Fletcher's approach of constructing a penalty function depending only on  $x$  becomes an excellent idea without the two shortcomings pointed out by Glad and Polak (cf. Section 3).

Moreover, as an extension of Fletcher's idea, we propose to construct a penalty function  $L(x(y), y, \alpha)$  depending only on the Lagrangian multiplier  $y$  if (4.3) always has a unique solution  $x(y)$ . This approach is suitable for a quadratic programming problem whose objective function has a nonsingular Hessian (cf. Section 4).

It is interesting to note that  $L(x, y(x), \alpha)$  or  $L(x(y), y, \alpha)$  can also be derived as a reformulation of the Karush-Kuhn-Tucker conditions of (4.1). This was the approach used for deriving the exact penalty functions  $\Psi(x)$  and  $\Phi(y)$  for quadratic programs without knowing that they are actually  $L(x, y(x), \alpha)$  and  $L(x(y), y, \alpha)$  [24, 25] (cf. also Sections 3 and 4).

The paper is organized as follows. General properties of the augmented Lagrangian function  $L(x, y, \alpha)$  for two-sided constraints are given in Section 2. The differentiable exact penalty functions  $L(x, y(x), \alpha)$  and  $L(x(y), y, \alpha)$  are discussed in Sections 3 and 4, respectively. Final comments are included in Section 5.

Now we give some terminologies and notations used in the paper.

For simplicity, we use  $f'(x)$  to denote the gradient of  $f(x)$  (as a column vector) and use  $f''(x)$  to denote the Hessian of  $f(x)$ . A real-valued function  $f(x)$  on  $\mathbf{R}^n$  is said to be a quadratic spline, if the gradient  $f'(x)$  of  $f(x)$  is a piecewise linear mapping from  $\mathbf{R}^n$  to  $\mathbf{R}^n$ . That is, a quadratic spline is a continuously differentiable piecewise quadratic function. The 2-norm  $\|\cdot\|$  on  $\mathbf{R}^n$  is defined as  $\|x\| := (\sum_{i=1}^n x_i^2)^{\frac{1}{2}}$  and the 2-norm of an  $n \times n$  matrix

$B$  is defined as  $\|B\| := \sup\{\|Bx\| : x \in \mathbf{R}^n \text{ with } \|x\| = 1\}$ . The transpose of a matrix  $B$  or a vector  $x$  is denoted by  $B^T$  or  $x^T$ . For  $x, y \in \mathbf{R}^n$ ,  $x \leq y$  means  $x_i \leq y_i$  for  $1 \leq i \leq n$ , where  $x_i$  or  $y_i$  denotes the  $i$ -th component of  $x$  or  $y$ . Let  $(z)_i^l$  be the lower and upper truncation of  $z$  by  $l$  and  $u$ , respectively, whose  $i$ -th component is  $\max\{l, \min\{u, z_i\}\}$ . By convention,  $z_+$  is a vector whose  $i$ -th component is  $\max\{z_i, 0\}$ . A vector  $x^* \in \mathbf{R}^n$  is said to be a local solution of (4.1) if  $l \leq g(x^*) \leq u$  and there exists a positive constant  $\delta$  such that  $f(x) \geq f(x^*)$  whenever  $l \leq g(x) \leq u$  and  $\|x - x^*\| < \delta$ . A vector  $x^* \in \mathbf{R}^n$  is said to be a global solution of (4.1) if  $l \leq g(x^*) \leq u$  and  $f(x) \geq f(x^*)$  whenever  $l \leq g(x) \leq u$ . A mapping  $x = x(w)$  from  $\mathbf{R}^k$  to  $\mathbf{R}^n$  is said to be an open mapping if  $x(\cdot)$  maps open sets in  $\mathbf{R}^k$  to open sets in  $\mathbf{R}^n$  (i.e.,  $\{x(w) : w \in U\}$  is open whenever  $U$  is an open subset of  $\mathbf{R}^k$ ). It is not difficult to verify that  $x(\cdot)$  is an open mapping if and only if, for any  $w^*$  and  $\delta > 0$ , there exists a positive constant  $\epsilon$  such that

$$\{x : \|x - x(w^*)\| < \epsilon\} \subset \{x(w) : \|w - w^*\| < \delta\}.$$

A mapping  $x = x(w)$  from  $\mathbf{R}^k$  to  $\mathbf{R}^n$  is said to be onto if its range is  $\mathbf{R}^n$  (i.e., for any  $\bar{x} \in \mathbf{R}^n$ , there exists  $w \in \mathbf{R}^k$  such that  $x(w) = \bar{x}$ ). The following definition of exact penalty functions was commonly used in literature (cf. [15]) and was formally given by Di Pillo and Grippo [7].

**Definition 18** *Let  $F(x)$  be a function from  $\mathbf{R}^n$  to  $\mathbf{R}$ . Then  $F(x)$  is said to be an exact penalty function of (4.1) with respect to a subset  $\mathcal{D}$  of  $\mathbf{R}^n$ , if  $x^*$  is a local (or global) solution of (4.1) whenever  $x^* \in \mathcal{D}$  is a local (or global) minimizer of  $F(x)$ .*

## 4.2 Some Properties of the Augmented Lagrangian Function

Most properties of the augmented Lagrangian function  $L(x, y, \alpha)$  given in this section are well-known for equality and one-sided inequality constraints (cf. [3]). Even though Bertsekas used one multiplier for two-sided inequality constraints in design of numerical methods for solving constrained minimization problems, he did not explicitly use the formula (4.2) as the augmented Lagrangian function for two-sided constraints [3, 1, 2]. Therefore, we give

complete proofs of these well-known properties of the augmented Lagrangian function here.

First we want to reformulate explicitly the Karush-Kuhn-Tucker conditions of (4.1) as a system of nonlinear equations. Special cases of the following lemma were given by Mangasarian (Lemma 2.1 in [27]), Glad and Polak (Lemma 1 in [12]), and Li and Swetits (Lemma 2.2 in [25]).

**Lemma 19** *For any  $\alpha > 0$ ,  $(x, y)$  is a Karush-Kuhn-Tucker point of (4.1) if and only if*

$$f'(x) + g'(x)y = 0 \quad \text{and} \quad g(x) = (g(x) + \alpha y)_l^u. \quad (4.8)$$

**Proof.** For any given point  $(x, y)$ ,  $(x, y)$  is a Karush-Kuhn-Tucker point of (4.1) if and only if  $(x, y)$  satisfies the following conditions:  $f'(x) + g'(x)y = 0$  and

$$\begin{aligned} g_i(x) &= l_i & \text{if } y_i < 0, \\ g_i(x) &= u_i & \text{if } y_i > 0, \\ l_i &\leq g_i(x) \leq u_i & \text{if } y_i = 0. \end{aligned} \quad (4.9)$$

It is not difficult to verify that, for any given  $\alpha > 0$ , (4.9) is equivalent to the system of nonlinear equations:  $g(x) = (g(x) + \alpha y)_l^u$ . ■

The following property of the augmented Lagrangian function  $L(x, y, \alpha)$  is well-known and is fundamental for deriving exact penalty functions from  $L(x, y, \alpha)$ .

**Lemma 20** *For any  $\alpha > 0$ , if  $l \leq g(x) \leq u$  and  $y \in \mathbf{R}^m$ , then*

$$L(x, y, \alpha) \leq f(x). \quad (4.10)$$

*The equality in (4.10) holds if  $(x, y)$  is a Karush-Kuhn-Tucker point of (4.1).*

**Proof.** Let  $l \leq g(x) \leq u$ . If  $\frac{1}{\alpha}(g_i(x) - u_i) + y_i > 0$ , then

$$0 < \frac{1}{\alpha}(g_i(x) - u_i) + y_i \leq y_i$$

and

$$\frac{1}{\alpha}(l_i - g_i(x)) - y_i = \frac{1}{\alpha}(l_i - u_i) - \left( \frac{1}{\alpha}(g_i(x) - u_i) + y_i \right) < 0.$$

As a consequence, we have that

$$\begin{aligned} \left(\frac{1}{\alpha}(l_i - g_i(x)) - y_i\right)_+ \cdot \left(\frac{1}{\alpha}(g_i(x) - u_i) + y_i\right)_+ &= 0, \\ \left(\frac{1}{\alpha}(g_i(x) - u_i) + y_i\right)_+^2 + \left(\frac{1}{\alpha}(l_i - g_i(x)) - y_i\right)_+^2 &\leq y_i^2. \end{aligned} \quad (4.11)$$

Similarly, (4.11) holds if  $\frac{1}{\alpha}(l_i - g_i(x)) - y_i > 0$ . Moreover, (4.11) trivially holds if both  $\frac{1}{\alpha}(l_i - g_i(x)) - y_i \leq 0$  and  $\frac{1}{\alpha}(g_i(x) - u_i) + y_i \leq 0$ . Therefore, it follows from the inequality in (4.11) that

$$\begin{aligned} L(x, y, \alpha) &= f(x) + \frac{\alpha}{2} \sum_{i=1}^m \left(\frac{1}{\alpha}(g_i(x) - u_i) + y_i\right)_+^2 \\ &\quad + \frac{\alpha}{2} \sum_{i=1}^m \left(\frac{1}{\alpha}(l_i - g_i(x)) - y_i\right)_+^2 - \frac{\alpha}{2} \sum_{i=1}^m y_i^2 \leq f(x). \end{aligned}$$

It follows from the equality in (4.11) and  $\frac{1}{\alpha}(z)_+ = \left(\frac{1}{\alpha}z\right)_+$  that

$$\begin{aligned} &\frac{\alpha}{2} \left\| \left(\frac{1}{\alpha}(g(x) - u) + y\right)_+ \right\|^2 + \frac{\alpha}{2} \left\| \left(\frac{1}{\alpha}(l - g(x)) - y\right)_+ \right\|^2 \\ &= \frac{\alpha}{2} \left\| \left(\frac{1}{\alpha}(g(x) - u) + y\right)_+ - \left(\frac{1}{\alpha}(l - g(x)) - y\right)_+ \right\|^2 \\ &= \frac{1}{2\alpha} \left\| ((g(x) - u) + \alpha y)_+ - ((l - g(x)) - \alpha y)_+ \right\|^2. \end{aligned} \quad (4.12)$$

Since  $(z)_+^u = z + (l - z)_+ - (z - u)_+$ , for  $z = g(x) + \alpha y$ , we obtain

$$(g(x) + \alpha y)_+^u = g(x) + \alpha y + ((l - g(x)) - \alpha y)_+ - ((g(x) - u) + \alpha y)_+. \quad (4.13)$$

If  $(x, y)$  satisfies the Karush-Kuhn-Tucker conditions, by Lemma 19 and (4.13), we get

$$\alpha y = ((g(x) - u) + \alpha y)_+ - ((l - g(x)) - \alpha y)_+. \quad (4.14)$$

It follows from (4.12) and (4.14) that  $L(x, y, \alpha) = f(x)$ . ■

**Lemma 21** Suppose that  $x = x(w)$  and  $y = y(w)$  are differentiable mappings of  $w$  and  $\alpha > 0$ . Then

$$\begin{aligned} L'_w(x(w), y(w), \alpha) &= x'(w)(f'(x) + g'(x)y) \\ &\quad + \frac{1}{\alpha}(x'(w)g'(x) + \alpha y'(w))(g(x) - (g(x) + \alpha y(x))_+^u). \end{aligned} \quad (4.15)$$

Moreover, the following statements are true:

1. If  $(x(w), y(w))$  is a Karush-Kuhn-Tucker point of (4.1), then

$$L'_w(x(w), y(w), \alpha) = 0.$$

2. Suppose that  $x(w)$  is an open and onto mapping. If  $(x(w^*), y(w^*))$  is a Karush-Kuhn-Tucker point of (4.1) and  $w^*$  is a local (or global) minimizer of  $L(x(w), y(w), \alpha)$ , then  $x(w^*)$  is a local (or global) solution of (4.1).

**Proof.** By the chain rule, we have that

$$\begin{aligned} L'_w(x(w), y(w), \alpha) &= x'(w)f'(x) - \alpha y'(w)y \\ &\quad + \alpha \left( \frac{1}{\alpha} x'(w)g'(x) + y'(w) \right) \left( \frac{1}{\alpha} (g(x) - u) + y \right)_+ \\ &\quad - \alpha \left( \frac{1}{\alpha} x'(w)g'(x) + y'(w) \right) \left( \frac{1}{\alpha} (l - g(x)) - y \right)_+. \end{aligned} \quad (4.16)$$

By  $(z)_l^u = z + (l - z)_+ - (z - u)_+$  and  $\left(\frac{1}{\alpha} z\right)_+ = \frac{1}{\alpha} (z)_+$ , we obtain that, for  $z = g(x) + \alpha y$ ,

$$\begin{aligned} &\left( \frac{1}{\alpha} (g(x) - u) + y \right)_+ - \left( \frac{1}{\alpha} (l - g(x)) - y \right)_+ \\ &= \frac{1}{\alpha} (g(x) - (g(x) + \alpha y)_l^u + \alpha y). \end{aligned} \quad (4.17)$$

The formula (4.15) follows from (4.16) and (4.17).

If  $(x(w), y(w))$  is a Karush-Kuhn-Tucker point of (4.1), by Lemma 19, we have  $f'(x) + g'(x)y = 0$  and  $g(x) = (g(x) + \alpha y)_l^u$ . Thus, by (4.15),  $L'_w(x(w), y(w), \alpha) = 0$ .

Now suppose  $(x(w^*), y(w^*))$  is a Karush-Kuhn-Tucker point of (4.1). If  $w^*$  is a local (or global) minimizer of  $L(x(w), y(w), \alpha)$ , then there exists a positive constant  $\delta$  such that

$$L(x(w), y(w), \alpha) \geq L(x(w^*), y(w^*), \alpha) \quad \text{for } \|w - w^*\| < \delta. \quad (4.18)$$

Since  $x(w)$  is an open mapping, there exists a positive constant  $\epsilon$  such that

$$\{x \in \mathbf{R}^n : \|x - x(w^*)\| < \epsilon\} \subset \{x(w) : \|w - w^*\| < \delta\}. \quad (4.19)$$

Now, let  $\bar{x} \in \mathbf{R}^n$  be such that  $\|\bar{x} - x(w^*)\| < \epsilon$  and  $l \leq g(\bar{x}) \leq u$ . By (4.19), there is  $w$  such that  $x(w) = \bar{x}$  and  $\|w - w^*\| < \delta$ . By Lemma 20 and (4.18), we get

$$f(\bar{x}) = f(x(w)) \geq L(x(w), y(w), \alpha) \geq L(x(w^*), y(w^*), \alpha) = f(x(w^*)).$$

Thus,  $x(w^*)$  is a local solution of (4.1).

Finally assume that  $w^*$  is actually a global minimizer of  $L(x(w), y(w), \alpha)$ . For any  $\bar{x}$  with  $l \leq g(\bar{x}) \leq u$ , there is  $w$  such that  $\bar{x} = x(w)$ , because  $x(w)$  is an onto mapping. Thus,

$$f(\bar{x}) = f(x(w)) \geq L(x(w), y(w), \alpha) \geq L(x(w^*), y(w^*), \alpha) = f(x(w^*))$$

and  $x(w^*)$  is a global solution of (4.1). ■

It is interesting to note that one necessary condition for a differentiable penalty function  $F(x)$  being exact is that the Karush-Kuhn-Tucker conditions of (4.1) can be derived from the  $n$  equations  $F'(x) = 0$ . In general, this is possible only if the multiplier  $y$  is uniquely determined by  $x$ . Therefore, it becomes clear why the extended Mangasarian-Fromovitz constraint qualification is a standard condition for differentiable exact penalty functions. In general, one might not be able to derive the  $m$  equations  $g(x) = (g(x) + \alpha y(x))_l^u$  in the Karush-Kuhn-Tucker conditions from the  $n$  equations  $F'(x) = 0$  if  $m > n$ .

### 4.3 Exact Penalty Function in Primal Variables

In this section, we discuss the penalty functions  $L(x, y(x), \alpha)$  derived from Hestenes-Powell-Rockafellar's augmented Lagrangian function by using Fletcher's multiplier  $y(x)$ . We assume that  $g'(x)$  has rank  $m$  and

$$y(x) = -(g'(x)^T g'(x))^{-1} g'(x)^T f'(x).$$

Note that, for  $w \equiv x$ ,  $x'(w)$  is the identity matrix. The following lemma is an immediate consequence of Lemma 21.

**Lemma 22** For any  $x \in \mathbf{R}^n$  and  $y \in \mathbf{R}^m$ ,

$$\begin{aligned} L'_x(x, y(x), \alpha) &= f'(x) + g'(x)y(x) \\ &\quad + \frac{1}{\alpha}(g'(x) + \alpha y'(x))(g(x) - (g(x) + \alpha y(x))_l^u). \end{aligned} \tag{4.20}$$



**Lemma 23** *If  $g'(x)^T g'(x) + \alpha g'(x)^T y'(x)$  is nonsingular for  $x$  in a subset  $\mathcal{D}$  of  $\mathbf{R}^n$ , then  $L(x, y(x), \alpha)$  is an exact penalty function of (4.1) with respect to  $\mathcal{D}$ .*

**Proof.** By the definition of  $y(x)$ , we have

$$g'(x)^T (f'(x) + g'(x)y(x)) = 0. \quad (4.21)$$

It follows from (4.20) and (4.21) that

$$\begin{aligned} \frac{1}{\alpha} (g'(x)^T g'(x) + \alpha g'(x)^T y'(x)) (g(x) - (g(x) + \alpha y(x)))^\alpha \\ = g'(x)^T L'_x(x, y(x), \alpha). \end{aligned} \quad (4.22)$$

Suppose that  $x^* \in \mathcal{D}$  and  $x^*$  is a local (or global) minimizer of  $L(x, y(x), \alpha)$ . Then  $L'_x(x^*, y(x^*), \alpha) = 0$ . Since  $x^* \in \mathcal{D}$ ,  $g'(x^*)^T g'(x^*) + \alpha g'(x^*)^T y'(x^*)$  is nonsingular. By (4.22), we have

$$g(x^*) - (g(x^*) + \alpha y(x^*))^\alpha = 0. \quad (4.23)$$

From  $L'_x(x^*, y(x^*), \alpha) = 0$ , (4.23), and (4.20), we also have that  $f'(x^*) + g'(x^*)y(x^*) = 0$ . By Lemma 19,  $(x^*, y(x^*))$  is a Karush-Kuhn-Tucker point of (4.1).

Note that, for  $w \equiv x$ ,  $x(w) = w$  is an open and onto mapping from  $\mathbf{R}^n$  to  $\mathbf{R}^n$ . Thus, if  $(x^*, y(x^*), \alpha)$  is a Karush-Kuhn-Tucker point of (4.1), by Lemma 21,  $x^*$  is a local (or global) solution of (4.1). This proves that  $L(x, y(x), \alpha)$  is an exact penalty function of (4.1) with respect to  $\mathcal{D}$ . ■

**Theorem 24** *Suppose that  $\mathcal{D}$  is compact. Then there exists a positive constant  $\alpha^*$  such that, for  $0 < \alpha < \alpha^*$ ,  $L(x, y(x), \alpha)$  is an exact penalty function of (4.1) with respect to  $\mathcal{D}$ .*

**Proof.** Note that

$$\begin{aligned} g'(x)^T g'(x) \left( I_m + \alpha (g'(x)^T g'(x))^{-1} g'(x)^T y'(x) \right) \\ = g'(x)^T g'(x) + \alpha g'(x)^T y'(x), \end{aligned} \quad (4.24)$$

where  $I_m$  is the identity matrix of order  $m$ . Thus,  $g'(x)^T g'(x) + \alpha g'(x)^T y'(x)$  is nonsingular if and only if

$$I_m + \alpha (g'(x)^T g'(x))^{-1} g'(x)^T y'(x) \quad (4.25)$$

is a nonsingular matrix.

Since  $\mathcal{D}$  is compact and  $g'(x), y'(x)$  are continuous mappings, we obtain

$$\gamma := \max_{x \in \mathcal{D}} \left\| \left( g'(x)^T g'(x) \right)^{-1} g'(x)^T y'(x) \right\| < \infty.$$

Let  $\alpha^* = \frac{1}{\gamma}$ . Then it is easy to verify that the matrix given in (4.25) is nonsingular for  $0 < \alpha < \alpha^*$ . Theorem 24 follows from Lemma 23. ■

**Theorem 25** Suppose that  $g(x) = Ax$ , where  $A$  is an  $m \times n$  matrix with rank  $m$ . If there exists a positive constant  $\alpha$  such that

$$0 < \alpha \left\| (AA^T)^{-1} A f''(x) A^T (AA^T)^{-1} \right\| < 1 \quad (4.26)$$

for  $x$  in a subset  $\mathcal{D}$  of  $\mathbf{R}^n$ , then  $L(x, y(x), \alpha)$  is an exact penalty function of (4.1) with respect to  $\mathcal{D}$ .

**Proof.** Since  $g(x) = Ax$ , it is easy to see that

$$y(x) = -(AA^T)^{-1} A f'(x)$$

and

$$y'(x) = -f''(x) A^T (AA^T)^{-1}.$$

The matrix given in (4.25) can be rewritten as follows:

$$I_m - \alpha (AA^T)^{-1} A f''(x) A^T (AA^T)^{-1}. \quad (4.27)$$

If (4.26) holds for  $x \in \mathcal{D}$ , then the matrix given in (4.27) is nonsingular for  $x \in \mathcal{D}$ ; and it follows from (4.24) that  $g'(x)^T g'(x) + \alpha g'(x)^T y'(x)$  is nonsingular for  $x \in \mathcal{D}$ . Thus, by Lemma 23,  $L(x, y(x), \alpha)$  is an exact penalty function of (4.1) with respect to  $\mathcal{D}$ . ■

An immediate consequence of Theorem 25 is the following result about an exact penalty function for quadratic programs with linearly independent constraints:

$$\min_x \frac{1}{2} x^T M x - b^T x \quad \text{subject to } l \leq A x \leq u, \quad (4.28)$$

where  $A$  is an  $m \times n$  matrix with rank  $m$ ,  $M$  is an  $n \times n$  symmetric matrix, and  $b \in \mathbf{R}^n$ . In this special case,  $y(x) = (AA^T)^{-1}A(Mx - b)$  and the augmented Lagrangian function has the following form:

$$L(x, y, \alpha) := \frac{1}{2}x^T Mx - b^T x + \frac{\alpha}{2} \left\| \left( \frac{1}{\alpha}(Ax - u) + y \right)_+ \right\|^2 + \frac{\alpha}{2} \left\| \left( \frac{1}{\alpha}(l - Ax) - y \right)_+ \right\|^2 - \frac{\alpha}{2} \|y\|^2. \quad (4.29)$$

**Theorem 26** Suppose that  $0 < \alpha \|(AA^T)^{-1}AM A^T(AA^T)^{-1}\| < 1$ ,  $L(x, y, \alpha)$  is given by (4.29), and  $y(x) := (AA^T)^{-1}A(b - Mx)$ . Then  $L(x, y(x), \alpha)$  is an exact penalty function of (4.28) with respect to  $\mathbf{R}^n$ .

The above theorem is an attempt to extend the following unconstrained reformulation of a convex quadratic program with simple bound constraints.

**Proposition 27** Suppose that  $A$  is the identity matrix  $I_n$  ( $m = n$ ) and  $M$  is positive semidefinite. Let  $0 < \alpha \|M\| < 1$  and

$$\Psi(x) := \frac{1}{2}x^T(E - E^2)x - \alpha b^T E x + \frac{1}{2}\|(c - Ex)_+\|^2 + \frac{1}{2}\|(Ex - d)_+\|^2, \quad (4.30)$$

where  $E := I - \alpha M$ ,  $c := l - \alpha b$ , and  $d := u - \alpha b$ . Then  $\Psi(x)$  is a convex quadratic spline. Moreover,  $x^*$  is a minimizer of  $\Psi(x)$  if and only if  $x^*$  is a solution of the convex quadratic program (4.28).

It is easy to verify that  $\frac{1}{\alpha}\Psi(x) - \frac{\alpha}{2}\|b\|^2 = L(x, y(x), \alpha)$ . Therefore, the above proposition shows a very important property of the penalty function  $L(x, y(x), \alpha)$ : it preserves the convexity of the original quadratic programming problem. As an extension of the above proposition, we have the following equivalent unconstrained reformulation of a convex quadratic program with linearly independent constraints.

**Theorem 28** Suppose that  $0 < \alpha \|(AA^T)^{-1}AM A^T(AA^T)^{-1}\| < 1$ ,  $M$  is positive semidefinite,  $y(x) := (AA^T)^{-1}A(b - Mx)$ , and  $L(x, y, \alpha)$  is given by (4.29). Then  $L(x, y(x), \alpha)$  is a convex quadratic spline function. Moreover,  $x^*$  is a minimizer of  $L(x, y(x), \alpha)$  if and only if  $x^*$  is a solution of the convex quadratic program (4.28).

**Proof.** Since  $y'(x) = -MA^T(AA^T)^{-1}$ , it follows from Lemma 22 that

$$L'_x(x, y(x), \alpha) = Px + p - \frac{1}{\alpha}Q^T(Qx + q)_l^u, \quad (4.31)$$

where  $p := A^T(AA^T)^{-1}Ab - b$ ,  $q := \alpha(AA^T)^{-1}Ab$ ,

$$\begin{aligned} P &:= M + \frac{1}{\alpha}A^TA - A^T(AA^T)^{-1}AM - MA^T(AA^T)^{-1}A, \\ Q &:= A - \alpha(AA^T)^{-1}AM. \end{aligned} \quad (4.32)$$

We claim that  $P$  and  $P - \frac{1}{\alpha}Q^TQ$  are positive semidefinite.

In fact, since  $\alpha > 0$ , it suffices to prove that  $P - \frac{1}{\alpha}Q^TQ$  is positive semidefinite. By simple algebraic manipulations, we obtain

$$P - \frac{1}{\alpha}Q^TQ = M - \alpha MA(AA^T)^{-1}(AA^T)^{-1}A^TM.$$

Since  $M$  is positive semidefinite, there exists an  $n \times n$  matrix  $B$  such that  $M = B^TB$ . Thus,

$$P - \frac{1}{\alpha}Q^TQ = B^T(I_n - \alpha BA(AA^T)^{-1}(AA^T)^{-1}A^TB)B.$$

Therefore,  $P - \frac{1}{\alpha}Q^TQ$  is positive semidefinite, if the symmetric matrix

$$I_n - \alpha BA(AA^T)^{-1}(AA^T)^{-1}A^TB^T \quad (4.33)$$

is positive semidefinite. For any matrix  $D$ , it is easy to verify that  $D^TD$  and  $DD^T$  have the same set of nonzero eigenvalues by using singular value decomposition. Since the 2-norm of a symmetric positive semidefinite matrix is its largest eigenvalue, we have  $\|D^TD\| = \|DD^T\|$ . Let  $D := BA(AA^T)^{-1}$ . Then

$$\begin{aligned} &\|BA(AA^T)^{-1}(AA^T)^{-1}A^TB^T\| \\ &= \|(AA^T)^{-1}A^TB^TBA(AA^T)^{-1}\| \\ &= \|(AA^T)^{-1}A^TMA(AA^T)^{-1}\|. \end{aligned}$$

Since  $0 < \alpha \|(AA^T)^{-1}AM A^T(AA^T)^{-1}\| < 1$ , all eigenvalues of the symmetric matrix

$$\alpha BA(AA^T)^{-1}(AA^T)^{-1}A^TB^T$$

are less than 1. Therefore, all eigenvalues of the symmetric matrix given in (4.33) are positive. As a consequence, the matrix given in (4.33) is positive definite. This completes the proof that  $P - \frac{1}{\alpha}Q^TQ$  is positive semidefinite.

From the proof of Lemma 2.1 in [25] we know that  $L'_x(x, y(x), \alpha)$  is a monotone mapping. As a consequence,  $L(x, y(x), \alpha)$  is a convex function [28].

Since  $M$  is positive semidefinite,  $x^*$  is a solution of (4.28) if and only if  $(x^*, y(x^*))$  is a Karush-Kuhn-Tucker point of (4.28). Similarly,  $x^*$  is a minimizer of  $L(x, y(x), \alpha)$  if and only if  $L'_x(x^*, y(x^*), \alpha) = 0$ , since  $L(x, y(x), \alpha)$  is convex. By Lemma 21 (1) and Theorem 26,  $(x^*, y(x^*))$  is a Karush-Kuhn-Tucker point of (4.28) if and only if  $L'_x(x, y(x), \alpha) = 0$ . Therefore,  $x^*$  is a solution of (4.28) if and only if  $x^*$  is a minimizer of  $L(x, y(x), \alpha)$ . ■

The above theorem was proved first by Li and Swetits [25] when  $m = n$  and  $A$  is a nonsingular matrix (cf. Proposition 27). Note that, if  $A$  is a nonsingular matrix, then (4.28) can be reformulated as a quadratic program with simple bound constraints by using the substitution  $\bar{x} = Ax$ . For quadratic programs with simple bound constraints, we actually have the following stronger result [21].

**Proposition 29** *Suppose that  $A$  is the identity matrix  $I_n$  of order  $n$  and  $L(x, y, \alpha)$  is given by (4.29). Let  $0 < 2\alpha\|M\| < 1$ . Then  $x^*$  is a local solution (or an isolated local solution, or a global solution) of (4.28) if and only if  $x^*$  is a local minimizer (or an isolated local minimizer, or a global minimizer) of the quadratic spline  $L(x, b - Mx, \alpha)$ .*

Based on Theorem 28 and Proposition 29, we are intrigued by the possibility of proving the following conjecture.

**Conjecture 30** *Consider the following unconstrained minimization of a quadratic spline:*

$$\min_x L(x, (AA^T)^{-1}A(b - Mx), \alpha), \quad (4.34)$$

*where  $L(x, y, \alpha)$  is given by (4.29). There exists a positive constant  $\alpha$  such that (4.28) and (4.34) have the same set of local solutions, the same set of isolated local solutions, and the same set of global solutions.*

**Remark.** As we mentioned before, it is easy to reformulate (4.28) as a quadratic program with simple bound constraints if  $A$  is nonsingular ( $m = n$ ) by using the substitution  $\bar{x} = Ax$ . Therefore, by Theorem 28 and Proposition 29, the unknown part of the above conjecture is when  $m < n$  and  $M$  is not positive semidefinite.

## 4.4 Exact Penalty Function in Dual Variables

As we pointed out in Section 2, if the Lagrangian multiplier  $y$  can not be uniquely determined by  $x$ , then it is unlikely to find a differentiable exact penalty function depending only on  $x$ . In this section, we discuss the possibility of deriving an exact penalty function in the dual variable  $y$  from the augmented Lagrangian function  $L(x, y, \alpha)$ . We assume that  $g(x) = Ax$ , where  $A$  is any  $m \times n$  matrix, and the Jacobian  $f''(x)$  of  $f(x)$  is always nonsingular. Then (4.3) has a unique solution  $x(y)$  for any given  $y$  and

$$x'(y) = -Af''(x)^{-1}. \quad (4.35)$$

For  $w \equiv y$ ,  $y'(w)$  is the identity matrix  $I_m$ . For any given  $y$  and  $g'(x) = A^T$ , by (4.15) and (4.35),

$$\begin{aligned} L'_y(x(y), y, \alpha) &= x'(y)(f'(x) + g'(x)y) \\ &\quad + \frac{1}{\alpha}(x'(y)g'(x) + \alpha I_m)(g(x) - (g(x) + \alpha y)_l^u) \\ &= \frac{1}{\alpha}(\alpha I_m - Af''(x)^{-1}A^T)(g(x) - (g(x) + \alpha y)_l^u), \end{aligned} \quad (4.36)$$

since  $(x(y), y)$  satisfies (4.3). By Lemma 19,  $(x(y), y)$  is a Karush-Kuhn-Tucker point of (4.1) if and only if

$$g(x(y)) - (g(x(y)) + \alpha y)_l^u = 0. \quad (4.37)$$

When the matrix

$$\alpha I_m - Af''(x)^{-1}A^T \quad (4.38)$$

is nonsingular, by (4.36),  $(x(y), y)$  is a Karush-Kuhn-Tucker point of (4.1) if and only if  $L'_y(x(y), y, \alpha) = 0$ . In the following two theorems, we state conditions on  $\alpha$  which ensure that the matrix given in (4.38) is nonsingular.

**Theorem 31** For any compact set  $\mathcal{D}$  of  $\mathbf{R}^m$ , let

$$\alpha^* := \max_{y \in \mathcal{D}} \|A f''(x(y))^{-1} A^T\| < \infty.$$

Then, for  $\alpha > \alpha^*$  and  $y \in \mathcal{D}$ ,  $L'_y(x(y), y, \alpha) = 0$  if and only if  $(x(y), y)$  is a Karush-Kuhn-Tucker point of (4.1).

**Theorem 32** Suppose that there exists  $\alpha$  such that  $\alpha > \|A^T f''(x)^{-1} A\|$  for  $x$  in a subset  $\mathcal{D}$  of  $\mathbf{R}^n$ . Then, for  $y \in \mathcal{D}$ ,  $L'_y(x(y), y, \alpha) = 0$  if and only if  $(x(y), y)$  is a Karush-Kuhn-Tucker point of (4.1).

Now consider the following quadratic programming problem:

$$\min_x \frac{1}{2} x^T M x - b^T x \quad \text{subject to} \quad l \leq A x \leq u, \quad (4.39)$$

where  $A$  is any  $m \times n$  matrix,  $M$  is an  $n \times n$  nonsingular symmetric matrix, and  $b \in \mathbf{R}^n$ . The corresponding augmented Lagrangian function  $L(x, y, \alpha)$  has the form given in (4.29).

**Theorem 33** Let  $x(y) = M^{-1}(b - A^T y)$  and  $L(x, y, \alpha)$  be given by (4.29). Then, for  $\alpha > \|A^T M^{-1} A\|$ ,  $L'_y(x(y), y, \alpha) = 0$  if and only if  $(x(y), y)$  is a Karush-Kuhn-Tucker point of (4.1).

The above theorem is a special case of Theorem 32. Note that  $x(y) = M^{-1}(b - A^T y)$  is an open and onto mapping when  $A$  has rank  $n$ . In this case, by Theorem 33 and Lemma 21,  $x(y^*)$  is a local (or global) solution of (4.1), if  $y^*$  is a local (or global) minimizer of  $L(x(y), y, \alpha)$  with  $\alpha > \|A^T M^{-1} A\|$ . We can consider  $L(x(y), y, \alpha)$  as an exact penalty function in dual variables.

**Corollary 34** Suppose that  $x(y) = M^{-1}(b - A^T y)$ ,  $L(x, y, \alpha)$  is given by (4.29), and  $A$  has rank  $n$ . Then, for  $\alpha > \|A^T M^{-1} A\|$ ,  $x(y^*)$  is a local (or global) solution of (4.1) if  $y^*$  is a local (or global) minimizer of  $L(x(y), y, \alpha)$ .

The above result is an attempt to extend the following unconstrained reformulation of a strictly convex quadratic programming problem [24, 25].

**Proposition 35** *Suppose that  $M$  is positive definite and  $\alpha > \|A^T M^{-1} A\|$ . Then  $x^* := M^{-1}(b - A^T y^*)$  is a solution of (4.39) and  $(x^*, y^*)$  is a Karush-Tuhn-Tucker point of (4.39) if and only if  $y^*$  is a minimizer of the following convex quadratic spline:*

$$\Phi(y) := \frac{\alpha}{2} y^T B y - \frac{1}{2} \|(By - c)_+\|^2 - \frac{1}{2} \|(d - By)_+\|^2, \quad (4.40)$$

where  $B = \alpha I - AM^{-1}A^T$ ,  $c := \alpha AM^{-1}b - l$ , and  $d := \alpha AM^{-1}b - u$ .

The penalty function  $\Phi(y)$  was introduced based on a reformulation of the Karush-Kuhn-Tucker conditions for (4.39) [25]. In fact, one can easily verify that

$$L(M^{-1}(b - A^T y), y, \alpha) = \frac{1}{\alpha} \Phi(y) - \frac{1}{2} b^T M^{-1} b. \quad (4.41)$$

Based on Proposition 35 and Corollary 34, it is reasonable to believe that (4.39) is equivalent to the unconstrained minimization of the quadratic spline  $L(M^{-1}(b - A^T y), y, \alpha)$ .

**Conjecture 36** *Consider the following unconstrained minimization problem:*

$$\min_y L(M^{-1}(b - A^T y), y, \alpha), \quad (4.42)$$

where  $L(x, y, \alpha)$  is the corresponding augmented Lagrangian function of (4.39) given in (4.29). Then there exists a positive constant  $\alpha$  such that  $y^*$  is a local solution (or global solution) of (4.42) if and only if  $x^* := M^{-1}(b - A^T y^*)$  is a local solution (or global solution) of (4.39) and  $(x^*, y^*)$  is a Karush-Kuhn-Tucker point of (4.39).

**Remark.** For the dual unconstrained reformulation, there might be many dual solutions  $y^*$  corresponding to one primal solution  $x^*$ . Therefore, it is impossible to establish correspondence between isolated local solutions. The conjecture is true if  $M$  is a positive definite symmetric matrix [25] (cf. Proposition 35).



## 4.5 Comments

With a differentiable exact penalty function, one can use various unconstrained minimization techniques to solve the original constrained minimization problem. See [3, 4, 5, 6, 7, 8, 9, 13, 14, 16, 19, 20, 23, 24, 25, 26] for some recent works as well as references on the subject.

It is interesting to note that, when  $(x(w), y(w))$  satisfies (4.3), one way to reformulate the Karush-Kuhn-Tucker conditions (4.8) is to find a function  $F(w)$  such that

$$F'(w) = B(w)(g(x(w)) - (g(x(w)) + \alpha y(w))_+^u), \quad (4.43)$$

where  $B(w)$  is a nonsingular matrix. Then  $(x(w), y(w))$  is a Karush-Kuhn-Tucker point of (4.1) if and only if  $F'(w) = 0$ . If the original constrained minimization problem is convex, then it is equivalent to the unconstrained minimization of the penalty function  $F(w)$ . The exact penalty functions  $\Psi(x)$  and  $\Phi(y)$  for convex quadratic programs with simple bound constraints and strictly convex quadratic programs, respectively, were constructed based on this approach [24, 25]. Later, Li proved that  $\Psi(x)$  is a strongly exact penalty function for any quadratic program with simple bound constraints [21]. Note that, by Lemma 21,  $F(w) := L(x(w), y(w), \alpha)$  satisfies (4.32) if  $(x(w), y(w))$  is a solution of (4.3). The differentiable exact penalty functions  $\Psi(x)$  and  $\Phi(y)$  can actually be derived from the augmented Lagrangian function by eliminating either  $x$  or  $y$  from the equation (4.3) (cf. Sections 3 and 4).

However, when the differentiable convex quadratic piecewise penalty functions either in the primal variable  $x$  or in the dual variable  $y$  were derived based on the reformulation (4.43) of the Karush-Kuhn-Tucker conditions, the special structures of the involved quadratic program seem to be crucial for the construction of these penalty functions. The author was asked, by P. Tseng, J.-S. Pang, and N. Gould in different circumstances, whether or not these penalty functions are related to differentiable penalty functions given by Di Pillo and Grippo or augmented Lagrangian functions. Our study reveals that the penalty functions derived for convex quadratic programs can actually be derived from Hestenes-Powell-Rockafellar's augmented Lagrangian function for two-sided constraints by eliminating either  $x$  or  $y$  by using the linear equation in the Karush-Kuhn-Tucker conditions. Furthermore, our results show that Fletcher's multiplier function  $y(x)$  (if works) produces an

exact penalty function for a quadratic programming problem on  $\mathbf{R}^n$  and the penalty parameter can be easily determined by the 2-norm of the matrix  $(AA^T)^{-1}AM A^T(AA^T)^{-1}$ .

The penalty function  $L(x(y), y, \alpha)$  seems to be extremely useful when one has to solve a sequence of separable strictly convex quadratic programming problems which are closely related to one another:

$$\min_x \frac{1}{2}x^T x - x^T b^k \text{ subject to } l^k \leq Ax \leq u^k, \quad (4.44)$$

where  $A$  has rank  $m$ . Let  $F_k(y) := L_k(b^k - A^T y, y, \alpha)$  be the corresponding differentiable piecewise quadratic exact penalty function of (4.44) as given in Theorem 33. In this special case, for  $\alpha > \|A^T A\|$ ,  $F_k(y)$  is actually a strictly convex quadratic spline (cf. [25]). In order to find the unique minimizer  $y^{k+1}$  of  $F_k(y)$ , we can use  $y^k$  as the initial guess. If  $b^k, l^k, u^k$  are small perturbations of  $b^{k-1}, l^{k-1}, u^{k-1}$ , respectively, then  $y^{k+1}$  should be close to  $y^k$  and it is easy to find  $y^{k+1}$  by starting from  $y^k$ . A data smoothing technique for piecewise convex/concave curves was developed based on the efficiency of a Newton method for finding the unique minimizer of  $F_k(y)$  starting from  $y^k$  (cf. [22]).

# Bibliography

- [1] D. P. Bertsekas, Multiplier methods: A survey, *Automatica-J. IFAC*, 12 (1976), 133-145.
- [2] D. P. Bertsekas, Approximation procedures based on the method of multipliers, *J. Optim. Theory Appl.*, 23 (1977), 487-510.
- [3] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.
- [4] J. V. Burke, An exact penalization viewpoint of constrained optimization, *SIAM J. Control Optim.*, 29 (1991), 968-998.
- [5] G. Contaldi, G. Di Pillo, and S. Lucidi, A continuously differentiable exact penalty function for nonlinear programming problems with unbounded feasible set, *Operations Research Letters*, 14 (1993), 153-161.
- [6] G. Di Pillo, F. Facchinei, and L. Grippo, An RQP algorithm using a differentiable exact penalty function for inequality constrained problems, *Math. Programming*, 55 (1992), Ser. A, 49-68.
- [7] G. Di Pillo and L. Grippo, Exact penalty functions in constrained optimization, *SIAM J. Control Optim.*, 27 (1989), 1333-1360.
- [8] F. Facchinei, Exact penalty functions and Lagrange multipliers, *Optimization*, 22 (1991), 579-606.
- [9] F. Facchinei and S. Lucidi, A class of penalty functions for optimization problems with bound constraints, *Optimization*, 26 (1992), 239-259.

- [10] R. Fletcher, A class of methods for nonlinear programming with termination and convergence properties, in "Integer and Nonlinear Programming", Edited by J. Abadie, pp. 157-173. North-Holland Publication, Amsterdam, 1970.
- [11] R. Fletcher, An exact penalty function for nonlinear programming with inequality constraints, *Math. Programming*, 5 (1973), 129-150.
- [12] T. Glad and E. Polak, A multiplier method with automatic limitation of penalty growth, *Math. Programming*, 17 (1979), 140-155.
- [13] L. Grippo and S. Lucidi, A differentiable exact penalty function for bound constrained quadratic programming problems, *Optimization*, 22 (1991), 557-578.
- [14] L. Grippo and S. Lucidi, On the solution of a class of quadratic programs using a differentiable exact penalty function, in "System Modelling and Optimization (Leipzig, 1989)", H. J. Sebastian and K. Tammer eds., pp. 764-773. Lecture Notes in Control and Inform. Sci., 143, Springer, Berlin, 1990.
- [15] S. P. Han and O. L. Mangasarian, Exact penalty functions in nonlinear programming, *Math. Programming*, 17 (1979), 251-269.
- [16] S. P. Han and O. L. Mangasarian, A dual differentiable exact penalty function, *Math. Programming*, 25 (1983), 293-306.
- [17] M. R. Hestenes, Multiplier and gradient methods, in "Computing Methods in Optimization Problems - 2", Edited by L. A. Zadeh, L. W. Neustadt, and A. V. Balakrishnan, pp. 143-164. Academic Press, Newt York, 1969.
- [18] M. R. Hestenes, Multiplier and gradient methods, *J. Optim. Theory Appl.*, 4 (1969), 303-320.
- [19] W. Li, Linearly convergent descent methods for unconstrained minimization of a convex quadratic spline, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, March 1993, (to appear in *J. Optim. Theory Appl.*).

- [20] W. Li, A conjugate gradient method for strictly convex quadratic programs with simple bound constraints, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, February 1994.
- [21] W. Li, A differentiable piecewise quadratic exact penalty function for quadratic programs with simple bound constraints, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, March 1994.
- [22] W. Li, D. Naik, and J. Swetits, An ideal data smoother for nonparametric regression, Technical Report TR92-12, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, Oct. 1992. (to appear in SIAM J. Scientific Statist. Computing)
- [23] W. Li, P. Pardalos, and C.-G. Han, Gauss-Seidel method for least distance problems, J. Optim. Theory Appl., 75 (1992), 487-500.
- [24] W. Li and J. Swetits, A Newton method for convex regression, data smoothing, and quadratic programming with bounded constraints, SIAM J. Optim., 3 (1993), 466-488.
- [25] W. Li and J. Swetits, A new algorithm for strictly convex quadratic programs, Technical Report TR92-1, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529, Nov. 1992. (to appear in SIAM J. Optim.)
- [26] S. Lucidi, New results on a continuously differentiable exact penalty function, SIAM J. Optim., 2 (1992), 558-574.
- [27] O. L. Mangasarian, Solution of symmetric linear complementarity problems by iterative methods, J. Optim. Theory Appl., 22 (1977), 465-485.
- [28] J. M. Ortega and W. C. Rheinboldt, *Iterative Solutions of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.
- [29] M. J. D. Powell, A method for nonlinear constraints in minimization problems, in "Optimization", Edited by R. Fletcher, pp. 283-298. Academic Press, New York, 1969.

- [30] R. T. Rockafellar, New applications of duality in convex programming, Proceedings of the Fourth Conference on Probability Theory, Brasov, Romania, 1971, pp. 73-81.
- [31] R. T. Rockafellar, The multiplier method of Hestenes and Powell applied to convex programming, J. Optim. Theory Appl., 12 (1973), 555-562.

